PROJECTE FINAL DE CARRERA

DIPLOMATURA D´ESTADÍSTICA

ISAX
(Interface for Seasonal Adjustment with X-12-ARIMA)

Alumna: Noemí Moltó Paz

Director: Albert Prat Bartés

ANEXO II(II) **Octubre 1997**

FACULTAT DE MATEMÀTIQUES I ESTADÍSTICA

ANEXO II

1.1.13. Xcheck.frm

```
VERSION 2.00
Begin Form Check
  BackColor     = &H00C0C0C0&
  BorderStyle   = 0 'None
  ClientHeight  = 4215
  ClientLeft    = 60
  ClientTop     = 2040
  ClientWidth   = 9060
  ControlBox    = 0 'False
  Height        = 4620
  HelpContextID = 3
  Left          = 0
  LinkTopic     = "Form8"
  MaxButton     = 0 'False
  MinButton     = 0 'False
  ScaleHeight   = 4215
  ScaleWidth    = 9060
  Top           = 1695
  Width         = 9180
  Begin SSPanel Panel3DCheck
    BackColor     = &H00C0C0C0&
    Caption       = " "
    Font3D        = 0 'None
    Height        = 4245
    Left          = 0
    TabIndex      = 1
    Top           = 0
    Width         = 9255
    Begin SSCommand Command3DCheck
      BevelWidth    = 1
      Caption       = "&OK"
      Font3D        = 0 'None
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 375
      HelpContextID = 3
      Left          = 7800
      TabIndex      = 0
      Top           = 3240
      Width         = 1095
    End
    Begin SSFrame Frame3DCheck
      Caption       = "Storage:"
      Font3D        = 0 'None
      ForeColor     = &H00000000&
      Height        = 1335
      Index         = 1
      Left          = 720
      TabIndex      = 5
      Top           = 2400
      Width         = 3495
      Begin SSCheck Check3DCheckData
        Caption       = "Partial autocorrelation function"
        Font3D        = 0 'None
        FontBold      = 0 'False
        FontItalic    = 0 'False
        FontName      = "MS Sans Serif"
        FontSize      = 8.25
        FontStrikethru = 0 'False
        FontUnderline = 0 'False
        Height        = 195
        HelpContextID = 300
        Index         = 1
        Left          = 240
        TabIndex      = 7
        Top           = 840
        Value         = -1 'True
```

```
      Width          = 3135
    End
    Begin SSCheck Check3DCheckData
      Caption        = "Autocorrelation function"
      Font3D         = 0 'None
      FontBold       = 0 'False
      FontItalic     = 0 'False
      FontName       = "MS Sans Serif"
      FontSize       = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height         = 255
      HelpContextID  = 300
      Index          = 0
      Left           = 240
      TabIndex       = 6
      Top            = 360
      Value          = -1 'True
      Width          = 2415
    End
  End
  Begin SSFrame Frame3DCheck
    Caption      = "ACF / PACF:"
    Font3D       = 0 'None
    ForeColor    = &H00000000&
    Height       = 1455
    Index        = 0
    Left         = 720
    TabIndex     = 2
    Top          = 480
    Width        = 3495
    Begin SpinButton SpinCheckMaxlag
      Height     = 375
      Left       = 2760
      Top        = 600
      Width      = 255
    End
    Begin TextBox TextCheckMaxlag
      Height     = 285
      HelpContextID = 301
      Left       = 2160
      TabIndex   = 3
      Text       = "36"
      Top        = 600
      Width      = 615
    End
    Begin Label Label
      BackStyle  = 0 'Transparent
      Caption    = " Number of flags:"
      FontBold   = 0 'False
      FontItalic = 0 'False
      FontName   = "MS Sans Serif"
      FontSize   = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height     = 255
      Index      = 0
      Left       = 360
      TabIndex   = 4
      Top        = 600
      Width      = 1575
    End
  End
  Begin Label Label
    AutoSize   = -1 'True
    BackStyle  = 0 'Transparent
    Caption    = "1/1"
    Height     = 195
    Index      = 4
    Left       = 8700
    TabIndex   = 8
    Top        = 120
    Width      = 315
```

```
    End
  End
End
Option Explicit

Sub cargar_check ()
Dim X%

  'For x% = 0 To 1
  '   Frame3DCheck(x%) = (getvar("check", "frame" & x%))
  'Next x%
  TextCheckMaxlag = (getvar("check", "textmaxlag"))
  'SpinCheckMaxlag = Val(getvar("check", "spinmaxlag"))
  For X% = 0 To 1
    Check3DCheckData(X%) = Val(getvar("check", "save" & X%))
  Next X%

End Sub

Sub command3dcheck_click ()
Dim X%

  'Call cancel_button
  'GUARDAR
  checkcargar(11) = "yes"
  Call setvar("check", "cargar", checkcargar(11))
  For X% = 0 To 1
    Call setvar("check", "frame" & X%, Frame3DCheck(X%))
  Next X%
  Call setvar("check", "textmaxlag", TextCheckMaxlag)
  'Call setvar("check", "spinmaxlag", spincheckmaxlag)
  For X% = 0 To 1
    Call setvar("check", "save" & X%, Check3DCheckData(X%))
  Next X%

End Sub

Sub Form_Load ()

 'If checkcargar(11) = "yes" Then
   Call cargar_check
 'End If

 Call command3dcheck_click

End Sub

Sub SpinCheckMaxlag_SpinDown ()

If Val(TextCheckMaxlag.Text) > 0 Then
  TextCheckMaxlag.Text = TextCheckMaxlag.Text - 1
End If

End Sub

Sub SpinCheckMaxlag_SpinUp ()

'no hay número máximo de retardos?????
TextCheckMaxlag.Text = TextCheckMaxlag.Text + 1

End Sub

Sub TextCheckMaxlag_KeyPress (keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextCheckMaxlag_LostFocus ()

  If Val(TextCheckMaxlag.Text) < 0 Then
    'mensaje de error
    response% = MsgBox("It must be a positive number", 48, title$)
```

```
        TextCheckMaxlag.SetFocus
    End If
'mirar si hay nº máx. de retardos o no !!!!!!

End Sub
```

1.1.14. Xforecast.frm

```
VERSION 2.00
Begin Form Forecast
  BackColor      = &H00C0C0C0&
  BorderStyle    = 0 'None
  ClientHeight   = 4230
  ClientLeft     = 60
  ClientTop      = 1425
  ClientWidth    = 9060
  ControlBox     = 0 'False
  Height         = 4635
  HelpContextID  = 6
  Left           = 0
  LinkTopic      = "Form9"
  MaxButton      = 0 'False
  MinButton      = 0 'False
  ScaleHeight    = 4230
  ScaleWidth     = 9060
  Top            = 1080
  Width          = 9180
  Begin SSPanel Panel3DForecast
    BackColor      = &H00C0C0C0&
    Caption        = " "
    Font3D         = 0 'None
    Height         = 4250
    Left           = 0
    TabIndex       = 1
    Top            = 0
    Width          = 9255
    Begin SSCommand Command3DForecast
      BevelWidth     = 1
      Caption        = "&OK"
      Font3D         = 0 'None
      FontBold       = 0 'False
      FontItalic     = 0 'False
      FontName       = "MS Sans Serif"
      FontSize       = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height         = 375
      HelpContextID  = 6
      Left           = 7860
      TabIndex       = 0
      Top            = 3240
      Width          = 1095
    End
    Begin SSFrame Frame3DForecast
      Caption        = "Storage:"
      Font3D         = 0 'None
      ForeColor      = &H00000000&
      Height         = 1455
      Left           = 360
      TabIndex       = 6
      Top            = 2400
      Width          = 5835
      Begin SSCheck Check3DForecast
        Caption        = "Forecasts on the original scale with upper and lower limits"
        Font3D         = 0 'None
        FontBold       = 0 'False
        FontItalic     = 0 'False
        FontName       = "MS Sans Serif"
        FontSize       = 8.25
        FontStrikethru = 0 'False
        FontUnderline  = 0 'False
        Height         = 195
        HelpContextID  = 600
        Index          = 2
        Left           = 240
        TabIndex       = 9
        Top            = 1080
        Value          = -1 'True
        Width          = 4395
```

```
  End
  Begin SSCheck Check3DForecast
    Caption       = "Forecast error variances on the transformated scale"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    HelpContextID = 600
    Index         = 1
    Left          = 240
    TabIndex      = 8
    Top           = 720
    Value         = -1 'True
    Width         = 4035
  End
  Begin SSCheck Check3DForecast
    Caption       = "Forecasts on the transformated scale with standard errors"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    HelpContextID = 600
    Index         = 0
    Left          = 240
    TabIndex      = 7
    Top           = 360
    Value         = -1 'True
    Width         = 4515
  End
End
Begin SpinButton SpinForecast
  Height        = 375
  Index         = 2
  Left          = 3900
  Top           = 1920
  Width         = 255
End
Begin TextBox TextForecast
  Height        = 285
  HelpContextID = 604
  Index         = 2
  Left          = 3360
  TabIndex      = 5
  Text          = "0"
  Top           = 1920
  Width         = 555
End
Begin TextBox TextForecast
  Height        = 285
  HelpContextID = 603
  Index         = 3
  Left          = 3360
  TabIndex      = 4
  Text          = "0.95"
  Top           = 1380
  Width         = 735
End
Begin SpinButton SpinForecast
  Height        = 375
  Index         = 1
  Left          = 3900
  Top           = 840
  Width         = 255
End
Begin SpinButton SpinForecast
```

```
      Height      = 375
      Index       = 0
      Left        = 3900
      Top         = 360
      Width       = 255
    End
    Begin TextBox TextForecast
      Height      = 285
      HelpContextID  = 602
      Index       = 1
      Left        = 3360
      TabIndex    = 3
      Text        = "0"
      Top         = 840
      Width       = 555
    End
    Begin TextBox TextForecast
      Height      = 285
      HelpContextID  = 601
      Index       = 0
      Left        = 3360
      TabIndex    = 2
      Top         = 360
      Width       = 555
    End
    Begin Label Label
      AutoSize    = -1 'True
      BackStyle   = 0 'Transparent
      Caption     = "1/1"
      Height      = 195
      Index       = 4
      Left        = 8700
      TabIndex    = 14
      Top         = 120
      Width       = 315
    End
    Begin Label Label
      BackStyle   = 0 'Transparent
      Caption     = "Number of obs. excluded before forecasting: "
      FontBold    = 0 'False
      FontItalic  = 0 'False
      FontName    = "MS Sans Serif"
      FontSize    = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height      = 435
      Index       = 1
      Left        = 420
      TabIndex    = 13
      Top         = 1860
      Width       = 2655
    End
    Begin Label Label
      BackStyle   = 0 'Transparent
      Caption     = "Probabilty of prediction intervals:"
      FontBold    = 0 'False
      FontItalic  = 0 'False
      FontName    = "MS Sans Serif"
      FontSize    = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height      = 195
      Index       = 98
      Left        = 420
      TabIndex    = 12
      Top         = 1380
      Width       = 2475
    End
    Begin Label Label
      BackStyle   = 0 'Transparent
      Caption     = "Number of backcasts:"
      FontBold    = 0 'False
      FontItalic  = 0 'False
```

```
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 195
      Index         = 87
      Left          = 420
      TabIndex      = 11
      Top           = 840
      Width         = 2115
    End
    Begin Label Label
      BackStyle     = 0 'Transparent
      Caption       = "Number of forecasts:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 195
      Index         = 76
      Left          = 420
      TabIndex      = 10
      Top           = 360
      Width         = 1635
    End
  End
End
Option Explicit

Sub cargar_forecast ()
Dim X%

  For X% = 0 To 3
    TextForecast(X%) = (getvar("forecast", "textforecast" & X%))
  Next X%
  'For x% = 0 To 2
    'SpinForecast(x%) = Val(getvar("forecast", "spinforecast" & x%))
  'Next x%
  'Frame3DForecast = (getvar("forecast", "frame"))
  For X% = 0 To 2
    Check3DForecast(X%) = Val(getvar("forecast", "save" & X%))
  Next X%

End Sub

Sub command3dforecast_click ()
Dim X%

  'Call cancel_button
  'GUARDAR
  checkcargar(12) = "yes"
  Call setvar("forecast", "cargar", checkcargar(12))
  For X% = 0 To 3
    Call setvar("forecast", "textforecast" & X%, TextForecast(X%))
  Next X%
  'For x% = 0 To 2
    'Call setvar("forecast", "spinforecast" & x%, spinforecast(x%))
  'Next x%
  Call setvar("forecast", "frame", Frame3DForecast)
  For X% = 0 To 2
    Call setvar("forecast", "save" & X%, Check3DForecast(X%))
  Next X%

End Sub

Sub Form_Load ()

  If checkcargar(12) = "yes" Then
    Call cargar_forecast
  * Else
  *   inicializar_forecast
```

```
  End If

  Call command3dforecast_click

End Sub

Sub SpinForecast_SpinDown (Index As Integer)

If Val(TextForecast(Index).Text) > 0 Then
   TextForecast(Index).Text = TextForecast(Index).Text - 1
End If

End Sub

Sub SpinForecast_SpinUp (Index As Integer)

Select Case Index
   Case 0 'maxlead
      If TextForecast(0).Text < 60 Then
         TextForecast(0).Text = TextForecast(0).Text + 1
      End If
   Case 1 'maxback
      If TextForecast(1).Text < 60 Then
         TextForecast(1).Text = TextForecast(1).Text + 1
      End If
   Case 2 'exclude
      'mirar el nº máx. de valores excluídos
      TextForecast(2).Text = TextForecast(2).Text + 1
End Select

End Sub

Sub TextForecast_KeyPress (Index As Integer, keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextForecast_LostFocus (Index As Integer)

Select Case Index
   Case 0 'maxlead
      If Val(TextForecast(0).Text) < 0 Then
         'mensaje de error
         response% = MsgBox("It must be a number from 0 to 60", 48, title$)
         TextForecast(0).SetFocus
      End If
      If Val(TextForecast(0).Text) > 60 Then
         'mensaje de error
         response% = MsgBox("It must be a number from 0 to 60", 48, title$)
         TextForecast(0).SetFocus
      End If
   Case 1 'maxback
      If Val(TextForecast(1).Text) <= 0 Then
         'mensaje de error
         response% = MsgBox("It must be a number from 0 to 60", 48, title$)
         TextForecast(1).SetFocus
      End If
      If Val(TextForecast(1).Text) > 60 Then
         'mensaje de error
         response% = MsgBox("It must be a number from 0 to 60", 48, title$)
         TextForecast(1).SetFocus
      End If
   Case 2 'exclude
      If Val(TextForecast(2).Text) < 0 Then
         'mensaje de error
         response% = MsgBox("It must be a positive number", 48, title$)
         TextForecast(2).SetFocus
      End If
      'mirar el nº máx. de valores excluídos
   Case 3 'probability
      'mirar los limites inferior y superior ??????
      'mensaje de error
```

```
End Select

End Sub
```

1.1.15. Xregadjust.frm

```
VERSION 2.00
Begin Form Regadjust
  BackColor     = &H00C0C0C0&
  BorderStyle   = 0 'None
  ClientHeight  = 4215
  ClientLeft    = 105
  ClientTop     = 1695
  ClientWidth   = 9060
  ControlBox    = 0 'False
  Height        = 4620
  HelpContextID = 10
  Left          = 45
  LinkTopic     = "Form1"
  MaxButton     = 0 'False
  MinButton     = 0 'False
  ScaleHeight   = 4215
  ScaleWidth    = 9060
  Top           = 1350
  Width         = 9180
  Begin SSPanel Panel3DRegadjust
    Alignment   = 6 'Center - TOP
    BackColor   = &H00C0C0C0&
    Font3D      = 0 'None
    ForeColor   = &H000000FF&
    Height      = 4250
    Index       = 0
    Left        = 0
    TabIndex    = 1
    Top         = 0
    Width       = 9255
    Begin SSCommand Command3DRegadjust
      BevelWidth    = 1
      Caption       = "&Next"
      Font3D        = 0 'None
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      ForeColor     = &H00000000&
      Height        = 375
      HelpContextID = 10
      Index         = 0
      Left          = 7800
      TabIndex      = 0
      Top           = 3240
      Width         = 1095
    End
    Begin SSFrame Frame3DRegadjust
      Enabled       = 0 'False
      Font3D        = 0 'None
      ForeColor     = &H00000000&
      Height        = 2775
      Index         = 0
      Left          = 360
      TabIndex      = 4
      Top           = 720
      Width         = 6615
      Begin SSCheck Check3DRegadjustPrior
        Caption       = "All the user-defined regression variables"
        Font3D        = 0 'None
        FontBold      = 0 'False
        FontItalic    = 0 'False
        FontName      = "MS Sans Serif"
        FontSize      = 8.25
        FontStrikethru = 0 'False
        FontUnderline = 0 'False
        Height        = 195
        HelpContextID = 1006
        Index         = 9
```

```
    Left        = 180
    TabIndex    = 9
    Top         = 2280
    Width       = 3795
  End
  Begin SSCheck Check3DRegadjustPrior
    Caption     = "Constant level shift"
    Font3D      = 0 'None
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    HelpContextID = 1005
    Index       = 7
    Left        = 180
    TabIndex    = 8
    Top         = 1800
    Width       = 2175
  End
  Begin SSCheck Check3DRegadjustPrior
    Caption     = "Additive outlier"
    Font3D      = 0 'None
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    HelpContextID = 1004
    Index       = 6
    Left        = 180
    TabIndex    = 7
    Top         = 1320
    Width       = 1755
  End
  Begin SSCheck Check3DRegadjustPrior
    Caption     = "Holiday"
    Font3D      = 0 'None
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    Index       = 3
    Left        = 180
    TabIndex    = 6
    Top         = 840
    Width       = 975
  End
  Begin SSCheck Check3DRegadjustPrior
    Caption     = "Trading day with six trading-day contrast variables"
    Font3D      = 0 'None
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    HelpContextID = 1002
    Index       = 0
    Left        = 180
    TabIndex    = 5
    Top         = 360
    Width       = 4695
  End
End
```

```
  Begin SSCheck Check3DRegadjustPriorVar
    Font3D       = 0 'None
    Height       = 195
    HelpContextID  = 1001
    Left         = 360
    TabIndex     = 2
    Top          = 240
    Width        = 195
  End
  Begin Label Label
    AutoSize     = -1 'True
    BackStyle    = 0 'Transparent
    Caption      = "1/4"
    Height       = 195
    Index        = 3
    Left         = 8700
    TabIndex     = 37
    Top          = 120
    Width        = 315
  End
  Begin Label Label
    BackStyle    = 0 'Transparent
    Caption      = "Regression effects to adjust the series before seasonal adjustment:"
    Height       = 435
    Index        = 0
    Left         = 600
    TabIndex     = 3
    Top          = 240
    Width        = 3315
  End
End
Begin SSPanel Panel3DRegadjust
  Alignment    = 6 'Center - TOP
  BackColor    = &H00C0C0C0&
  Font3D       = 0 'None
  ForeColor    = &H000000FF&
  Height       = 4250
  Index        = 1
  Left         = 0
  TabIndex     = 10
  Top          = 0
  Width        = 9255
  Begin SSCommand Command3DRegadjust
    BevelWidth   = 1
    Caption      = "&Back"
    Font3D       = 0 'None
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline  = 0 'False
    ForeColor    = &H00000000&
    Height       = 375
    Index        = 4
    Left         = 7800
    TabIndex     = 33
    Top          = 3720
    Width        = 1095
  End
  Begin SSCommand Command3DRegadjust
    BevelWidth   = 1
    Caption      = "&Next"
    Font3D       = 0 'None
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline  = 0 'False
    ForeColor    = &H00000000&
    Height       = 375
    Index        = 1
```

```
  Left        = 7800
  TabIndex    = 18
  Top         = 3240
  Width       = 1095
End
Begin SSFrame Frame3DRegadjust
  Enabled     = 0  'False
  Font3D      = 0  'None
  ForeColor   = &H00000000&
  Height      = 2415
  Index       = 1
  Left        = 300
  TabIndex    = 13
  Top         = 840
  Width       = 6315
  Begin SSCheck Check3DRegadjustFinal
    Caption       = "All the user-defined regression variables"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 1006
    Index         = 6
    Left          = 240
    TabIndex      = 17
    Top           = 1800
    Width         = 3735
  End
  Begin SSCheck Check3DRegadjustFinal
    Caption       = "Constant level shift"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 1005
    Index         = 4
    Left          = 240
    TabIndex      = 16
    Top           = 1320
    Width         = 1935
  End
  Begin SSCheck Check3DRegadjustFinal
    Caption       = "Additive outlier"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 1004
    Index         = 3
    Left          = 240
    TabIndex      = 15
    Top           = 840
    Width         = 1635
  End
  Begin SSCheck Check3DRegadjustFinal
    Caption       = "Holiday"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
```

```
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height = 195
      HelpContextID = 1003
      Index = 0
      Left = 240
      TabIndex = 14
      Top = 360
      Width = 1635
    End
  End
  Begin SSCheck Check3DRegadjustFinalVar
    Font3D = 0 'None
    Height = 195
    HelpContextID = 1007
    Left = 300
    TabIndex = 11
    Top = 240
    Width = 195
  End
  Begin Label Label
    AutoSize = -1 'True
    BackStyle = 0 'Transparent
    Caption = "2/4"
    Height = 195
    Index = 4
    Left = 8700
    TabIndex = 38
    Top = 120
    Width = 315
  End
  Begin Label Label
    BackStyle = 0 'Transparent
    Caption = "Regression effects to be removed from the final seasonally adjusted series:"
    Height = 435
    Index = 1
    Left = 540
    TabIndex = 12
    Top = 240
    Width = 3375
  End
End
Begin SSPanel Panel3DRegadjust
  BackColor = &H00C0C0C0&
  Font3D = 0 'None
  ForeColor = &H000000FF&
  Height = 4255
  Index = 2
  Left = 0
  TabIndex = 30
  Top = 0
  Width = 9255
  Begin SSFrame Frame3DRegadjust
    Enabled = 0 'False
    Font3D = 0 'None
    ForeColor = &H00000000&
    Height = 2655
    Index = 3
    Left = 420
    TabIndex = 40
    Top = 480
    Width = 6015
    Begin ListBox ListChoices
      Height = 1785
      HelpContextID = 1008
      Left = 4800
      TabIndex = 46
      Top = 600
      Width = 1095
    End
    Begin ListBox ListDifferent
      Height = 1200
      HelpContextID = 1008
```

```
    Left        = 480
    TabIndex    = 44
    Top         = 1200
    Visible     = 0  'False
    Width       = 1095
  End
  Begin SSOption Option3Duser
    Caption     = "Different type for each user-defined variable"
    Font3D      = 0  'None
    FontBold    = 0  'False
    FontItalic  = 0  'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height      = 195
    HelpContextID = 1008
    Index       = 1
    Left        = 240
    TabIndex    = 42
    TabStop     = 0  'False
    Top         = 660
    Width       = 3435
  End
  Begin SSOption Option3Duser
    Caption     = "Same type for all the user-defined variables"
    Font3D      = 0  'None
    FontBold    = 0  'False
    FontItalic  = 0  'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height      = 195
    HelpContextID = 1008
    Index       = 0
    Left        = 240
    TabIndex    = 41
    Top         = 360
    Value       = -1  'True
    Width       = 3495
  End
  Begin Label Label
    BorderStyle = 1  'Fixed Single
    Height      = 255
    Index       = 8
    Left        = 3840
    TabIndex    = 47
    Top         = 360
    Width       = 855
  End
  Begin Label Label
    BackStyle   = 0  'Transparent
    Caption     = "Choices:"
    FontBold    = 0  'False
    FontItalic  = 0  'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height      = 255
    Index       = 7
    Left        = 4800
    TabIndex    = 45
    Top         = 360
    Width       = 735
  End
  Begin Label Label
    BackStyle   = 0  'Transparent
    Height      = 255
    Index       = 6
    Left        = 480
    TabIndex    = 43
```

```
    Top          = 960
    Visible      = 0  'False
    Width        = 2535
  End
End
Begin ComboBox ComboRegadjustAugmentusertd
  Height       = 300
  HelpContextID  = 1009
  Left         = 480
  Style        = 2  'Dropdown List
  TabIndex     = 19
  Top          = 3600
  Width        = 5175
End
Begin SSCommand Command3DRegadjust
  BevelWidth   = 1
  Caption      = "&Back"
  Font3D       = 0  'None
  FontBold     = 0  'False
  FontItalic   = 0  'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0  'False
  FontUnderline  = 0  'False
  ForeColor    = &H00000000&
  Height       = 375
  Index        = 5
  Left         = 7800
  TabIndex     = 34
  Top          = 3720
  Width        = 1095
End
Begin SSCheck Check3DUser
  Caption      = "Type of model-estimated effect for the user regression variables:"
  Font3D       = 0  'None
  Height       = 255
  HelpContextID  = 1008
  Left         = 420
  TabIndex     = 32
  Top          = 240
  Width        = 5835
End
Begin SSCommand Command3DRegadjust
  BevelWidth   = 1
  Caption      = "&Next"
  Font3D       = 0  'None
  FontBold     = 0  'False
  FontItalic   = 0  'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0  'False
  FontUnderline  = 0  'False
  ForeColor    = &H00000000&
  Height       = 375
  Index        = 2
  Left         = 7800
  TabIndex     = 31
  Top          = 3240
  Width        = 1095
End
Begin Label Label
  BackStyle    = 0  'Transparent
  Caption      = "Trading day adjustment factors:"
  Height       = 195
  Index        = 2
  Left         = 480
  TabIndex     = 20
  Top          = 3360
  Width        = 2835
End
Begin Label Label
  AutoSize     = -1  'True
  BackStyle    = 0  'Transparent
```

```
    Caption      = "3/4"
    Height       = 195
    Index        = 5
    Left         = 8700
    TabIndex     = 39
    Top          = 120
    Width        = 315
  End
End
Begin SSPanel Panel3DRegadjust
  Alignment      = 6 'Center - TOP
  BackColor      = &H00C0C0C0&
  Font3D         = 0 'None
  ForeColor      = &H000000FF&
  Height         = 4250
  Index          = 3
  Left           = 0
  TabIndex       = 21
  Top            = 0
  Width          = 9255
  Begin SSCommand Command3DRegadjust
    BevelWidth   = 1
    Caption      = "&Back"
    Font3D       = 0 'None
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    ForeColor    = &H00000000&
    Height       = 375
    Index        = 6
    Left         = 7800
    TabIndex     = 35
    Top          = 3720
    Width        = 1095
  End
  Begin SSFrame Frame3DRegadjust
    Caption      = "Storage:"
    Font3D       = 0 'None
    ForeColor    = &H00000000&
    Height       = 2295
    Index        = 2
    Left         = 360
    TabIndex     = 23
    Top          = 480
    Width        = 4455
    Begin SSCheck Check3DRegadjustSave
      Caption    = "Original series adjusted for regARIMA outliers"
      Font3D     = 0 'None
      FontBold   = 0 'False
      FontItalic = 0 'False
      FontName   = "MS Sans Serif"
      FontSize   = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height     = 195
      HelpContextID = 1000
      Index      = 5
      Left       = 240
      TabIndex   = 29
      Top        = 1860
      Width      = 3555
    End
    Begin SSCheck Check3DRegadjustSave
      Caption    = "Factors from user-defined regression variables"
      Font3D     = 0 'None
      FontBold   = 0 'False
      FontItalic = 0 'False
      FontName   = "MS Sans Serif"
      FontSize   = 8.25
      FontStrikethru = 0 'False
```

```
    FontUnderline  =  0  'False
    Height         =  195
    HelpContextID  =  1000
    Index          =  4
    Left           =  240
    TabIndex       =  28
    Top            =  1560
    Value          =  -1  'True
    Width          =  3555
  End
  Begin SSCheck Check3DRegadjustSave
    Caption        =  "RegARIMA holiday factors"
    Font3D         =  0  'None
    FontBold       =  0  'False
    FontItalic     =  0  'False
    FontName       =  "MS Sans Serif"
    FontSize       =  8.25
    FontStrikethru =  0  'False
    FontUnderline  =  0  'False
    Height         =  195
    HelpContextID  =  1000
    Index          =  3
    Left           =  240
    TabIndex       =  27
    Top            =  1260
    Value          =  -1  'True
    Width          =  2775
  End
  Begin SSCheck Check3DRegadjustSave
    Caption        =  "RegARIMA trading day factors"
    Font3D         =  0  'None
    FontBold       =  0  'False
    FontItalic     =  0  'False
    FontName       =  "MS Sans Serif"
    FontSize       =  8.25
    FontStrikethru =  0  'False
    FontUnderline  =  0  'False
    Height         =  195
    HelpContextID  =  1000
    Index          =  2
    Left           =  240
    TabIndex       =  26
    Top            =  960
    Value          =  -1  'True
    Width          =  2775
  End
  Begin SSCheck Check3DRegadjustSave
    Caption        =  "RegARIMA level change and ramp factors"
    Font3D         =  0  'None
    FontBold       =  0  'False
    FontItalic     =  0  'False
    FontName       =  "MS Sans Serif"
    FontSize       =  8.25
    FontStrikethru =  0  'False
    FontUnderline  =  0  'False
    Height         =  195
    HelpContextID  =  1000
    Index          =  1
    Left           =  240
    TabIndex       =  25
    Top            =  660
    Value          =  -1  'True
    Width          =  3375
  End
  Begin SSCheck Check3DRegadjustSave
    Caption        =  "RegARIMA additive outlier factors"
    Font3D         =  0  'None
    FontBold       =  0  'False
    FontItalic     =  0  'False
    FontName       =  "MS Sans Serif"
    FontSize       =  8.25
    FontStrikethru =  0  'False
    FontUnderline  =  0  'False
```

```
        Height      = 195
        HelpContextID  = 1000
        Index       = 0
        Left        = 240
        TabIndex    = 24
        Top         = 360
        Value       = -1 'True
        Width       = 2775
      End
    End
    Begin SSCommand Command3DRegadjust
      BevelWidth   = 1
      Caption      = "&OK"
      Font3D       = 0 'None
      FontBold     = 0 'False
      FontItalic   = 0 'False
      FontName     = "MS Sans Serif"
      FontSize     = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      ForeColor    = &H00000000&
      Height       = 375
      Index        = 3
      Left         = 7800
      TabIndex     = 22
      Top          = 3240
      Width        = 1095
    End
    Begin Label Label
      AutoSize     = -1 'True
      BackStyle    = 0 'Transparent
      Caption      = "4/4"
      Height       = 195
      Index        = 11
      Left         = 8700
      TabIndex     = 36
      Top          = 120
      Width        = 315
    End
  End
End
Option Explicit

Sub cargar_regadjust ()
Dim X%

 Check3DRegadjustPriorVar = Val(getvar("regadjust", "checkpriorvar"))
 'For X% = 0 To 2
 ' Frame3DRegadjust(X%) = (getvar("regadjust", "frame" & X%))
 'Next X%
 For X% = 0 To 9
  Select Case X% 'n
    Case 0, 3, 6, 7, 9 'n
      Check3DRegadjustPrior(X%) = Val(getvar("regadjust", "checkprior" & X%))
  End Select 'n
 Next X%
 Check3DRegadjustFinalVar = Val(getvar("regadjust", "checkfinalvar"))
 For X% = 0 To 6
  Select Case X% 'n
    Case 0, 3, 4, 6'n
      Check3DRegadjustFinal(X%) = Val(getvar("regadjust", "checkfinal" & X%))
  End Select 'n
 Next X%
 ComboRegadjustAugmentusertd.ListIndex = Val(getvar("regadjust", "augmentusertd"))
 'ComboRegadjustAugmentusertd = Val(getvar("regadjust", "augmentusertd"))
 For X% = 0 To 5
  Check3DRegadjustSave(X%) = Val(getvar("regadjust", "save" & X%))
 Next X%
 For X% = 0 To 11 'n
  'Select Case X% 'n
  '  Case 0, 1, 2, 3, 5, 6 'n
      listchoices.AddItem (getvar("regadjust", "listchoices" & X%))
  'End Select 'n
```

```
 Next X%
 check3duser = Val(getvar("regadjust", "checkuser"))
 For X% = 0 To 1
  option3duser(X%) = Val(getvar("regadjust", "optionuser" & X%))
 Next X%
 'label(6) = (getvar("regadjust", "label6"))
 label(6).Caption = Val(getvar("regression", "maxuser")) + 1 & " regression variables"
 label(8) = (getvar("regadjust", "label8"))
 maxuser% = Val(getvar("regadjust", "maxuser"))

 If (getvar("regadjust", "listdifferent0")) <> "" Then
  listdifferent.List(0) = (getvar("regadjust", "listdifferent0"))
 End If
 For X% = 1 To maxuser% - 1
  listdifferent.List(X%) = (getvar("regadjust", "listdifferent" & X%))
 Next X%

End Sub

Sub command3dregadjust_click (Index As Integer)
Dim X%, l1%, maxuser%
ReDim l2(20)

 Select Case Index
  Case 0 'next primer panel regadjust
   panel3dregadjust(1).ZOrder
  Case 1 'next segundo panel regadjust
   panel3dregadjust(2).ZOrder
  Case 2 'cancel segundo panel regadjust
   panel3dregadjust(3).ZOrder
  Case 3 'ok tercer panel de regadjust
   checkcargar(13) = "yes"
   Call setvar("regadjust", "cargar", checkcargar(13))
   Call setvar("regadjust", "checkpriorvar", Check3DRegadjustPriorVar)
   For X% = 0 To 2
    Call setvar("regadjust", "frame" & X%, frame3dregadjust(X%))
   Next X%
   For X% = 0 To 9
    Select Case X% 'n
      Case 0, 3, 6, 7, 9 'n
        Call setvar("regadjust", "checkprior" & X%, Check3DRegadjustPrior(X%))
    End Select 'n
   Next X%
   Call setvar("regadjust", "checkfinalvar", Check3DRegadjustFinalVar)
   For X% = 0 To 6
    Select Case X%'n
      Case 0, 3, 4, 6'n
        Call setvar("regadjust", "checkfinal" & X%, Check3DRegadjustFinal(X%))
    End Select 'n
   Next X%
   l1 = ComboRegadjustAugmentusertd.ListIndex
   Call setvar("regadjust", "augmentusertd", l1)
   'Call setvar("regadjust", "augmentusertd", comboregadjustaugmentusertd)
   For X% = 0 To 5
    Call setvar("regadjust", "save" & X%, Check3DRegadjustSave(X%))
   Next X%
   Call setvar("regadjust", "checkuser", check3duser)
   For X% = 0 To 1
    Call setvar("regadjust", "optionuser" & X%, option3duser(X%))
   Next X%
   Call setvar("regadjust", "label6", label(6))
   Call setvar("regadjust", "label8", label(8))
   maxuser% = listdifferent.ListCount
   Call setvar("regadjust", "maxuser", maxuser%)
   For X% = 0 To maxuser%
    l2(X%) = listdifferent.List(X%)
    Call setvar("regadjust", "listdifferent" & X%, l2(X%))
   Next X%
  Case 4 'back segundo panel
   panel3dregadjust(0).ZOrder
  Case 5 'back tercer panel
   panel3dregadjust(1).ZOrder
  Case 6 'back cuarto panel
```

```
    panel3dregadjust(2).ZOrder
  End Select
End Sub

Sub Check3DRegadjustFinalVar_Click (Value As Integer)
Dim i%

   If Value = False Then
   For i% = 0 To 6
      Select Case i% 'n
         Case 0, 3, 4, 6 'n
            Check3DRegadjustFinal(i%).Value = False
      End Select 'n
   Next i%
   End If
   frame3dregadjust(1).Enabled = Value

End Sub

Sub Check3DRegadjustPriorVar_Click (Value As Integer)
Dim i%

   If Value = False Then
   For i% = 0 To 9
      Select Case i% 'n
         Case 0, 3, 6, 7, 9 'n
            Check3DRegadjustPrior(i%).Value = False
      End Select 'n
   Next i%
   End If
   frame3dregadjust(0).Enabled = Value

End Sub

Sub Check3DUser_Click (Value As Integer)

   Select Case Value
      Case True
         If (getvar("regression", "maxuser")) <> "" And Val(getvar("regression", "maxuser")) >= 0 And Val(getvar("regression",
"checkuservar")) <> 0 Then
            frame3dregadjust(3).Enabled = True
            listdifferent.Enabled = True
            label(8).Enabled = True
            Else
               response = MsgBox("You don't have any user regression variable", 48, title)
               check3duser.Value = False
         End If
      Case False
         listdifferent.Enabled = False
         label(8).Enabled = False
         frame3dregadjust(3).Enabled = False
   End Select

End Sub

Sub Form_Load ()

   panel3dregadjust(0).ZOrder

   Call combo_regadjust

   'If checkcargar(13) = "yes" Then
    Call cargar_regadjust
   'End If

   Call command3dregadjust_click(3)

End Sub

Sub Label_Click (Index As Integer)
Dim i%

   Select Case Index
```

```
    Case 0
      Check3DRegadjustPriorVar.Value = Not (Check3DRegadjustPriorVar.Value)
    Case 1
      Check3DRegadjustFinalVar.Value = Not (Check3DRegadjustFinalVar.Value)
  End Select

  frame3dregadjust(0).Enabled = Check3DRegadjustPriorVar.Value
  frame3dregadjust(1).Enabled = Check3DRegadjustFinalVar.Value

End Sub

Sub ListChoices_DblClick ()

  If option3duser(0).Value = True Then
    If (getvar("regression", "maxuser")) <> "" And Val(getvar("regression", "maxuser")) >= 0 And Val(getvar("regression",
"checkuservar")) <> 0 Then
      label(8).Caption = listchoices.List(listchoices.ListIndex)
      Else
        response = MsgBox("You don't have any user regression variable", 48, title)
        check3duser.Value = False
        label(6).Visible = False
        label(8).Visible = False
        listdifferent.Visible = False
        listchoices.ListIndex = -1
    End If
  End If
  If option3duser(1).Value = True Then
    If (getvar("regression", "maxuser")) <> "" And Val(getvar("regression", "maxuser")) >= 0 And Val(getvar("regression",
"checkuservar")) <> 0 Then
      If listdifferent.ListCount <= Val(getvar("regression", "maxuser")) Then
        listdifferent.AddItem listchoices.List(listchoices.ListIndex)
        Else
          response = MsgBox("You only have " & Val(getvar("regression", "maxuser")) + 1 & " regression variables", 48, title)
      End If
      Else
        response = MsgBox("You don't have any user regression variable", 48, title)
        check3duser.Value = False
        label(6).Visible = False
        label(8).Visible = False
        listdifferent.Visible = False
        listchoices.ListIndex = -1
    End If
  End If

End Sub

Sub ListDifferent_KeyDown (keycode As Integer, Shift As Integer)

  Select Case keycode
    Case 46 'suprimir
      listdifferent.RemoveItem listdifferent.ListIndex
  End Select

End Sub

Sub Option3Duser_Click (Index As Integer, Value As Integer)

  Select Case Index
    Case 0 'same type
      Select Case Value
        Case True
          label(6).Visible = False
          listdifferent.Visible = False
          label(8).Visible = True
        Case False
          label(8).Visible = False
          label(6).Visible = True
          listdifferent.Visible = True
          If (getvar("regression", "maxuser")) <> "" And Val(getvar("regression", "maxuser")) >= 0 And Val(getvar("regression",
"checkuservar")) <> 0 Then
            label(6).Caption = Val(getvar("regression", "maxuser")) + 1 & " regression variables"
            Else
              label(6).Caption = "0 regression variables"
```

```
            End If
        End Select
    Case 1 'different type
        Select Case Value
            Case True
                label(6).Visible = True
                listdifferent.Visible = True
                label(8).Visible = False
                If (getvar("regression", "maxuser")) <> "" And Val(getvar("regression", "maxuser")) >= 0 And Val(getvar("regression",
"checkuservar")) <> 0 Then
                    label(6).Caption = Val(getvar("regression", "maxuser")) + 1 & " regression variables"
                    Else
                        label(6).Caption = "0 regression variables"
                End If
            Case False
                label(6).Visible = False
                listdifferent.Visible = False
                label(8).Visible = True
        End Select
End Select

End Sub
```

1.1.16. XSliding.frm

```
VERSION 2.00
Begin Form Sliding
  BackColor      = &H00C0C0C0&
  BorderStyle    = 0 'None
  ClientHeight   = 4215
  ClientLeft     = -75
  ClientTop      = 1830
  ClientWidth    = 9060
  ControlBox     = 0 'False
  Height         = 4620
  HelpContextID  = 13
  Left           = -135
  LinkTopic      = "Form2"
  MaxButton      = 0 'False
  MinButton      = 0 'False
  ScaleHeight    = 4215
  ScaleWidth     = 9060
  Top            = 1485
  Width          = 9180
  Begin SSPanel Panel3DSliding
    BackColor    = &H00C0C0C0&
    Font3D       = 0 'None
    Height       = 4250
    Index        = 0
    Left         = 0
    TabIndex     = 1
    Top          = 0
    Width        = 9255
    Begin ComboBox ComboSlidingFixmdl
      Height       = 300
      HelpContextID = 1307
      Left         = 5040
      Style        = 2 'Dropdown List
      TabIndex     = 12
      Top          = 2400
      Width        = 3975
    End
    Begin ComboBox ComboSlidingOutlier
      Height       = 300
      HelpContextID = 1306
      Left         = 5040
      Style        = 2 'Dropdown List
      TabIndex     = 34
      Top          = 1440
      Width        = 3975
    End
    Begin ComboBox ComboSlidingFixmissing
      Height       = 300
      HelpContextID = 1308
      Left         = 5040
      Style        = 2 'Dropdown List
      TabIndex     = 22
      Top          = 480
      Width        = 3975
    End
    Begin SSFrame Frame3DSliding
      Caption      = "Threshold values:"
      Font3D       = 0 'None
      ForeColor    = &H00000000&
      Height       = 2175
      Index        = 0
      Left         = 240
      TabIndex     = 14
      Top          = 1440
      Width        = 4575
      Begin TextBox TextSlidingCuttd
        Height       = 285
        HelpContextID = 1305
        Left         = 3600
        TabIndex     = 20
        Text         = "2.0"
```

```
    Top             = 1680
    Width           = 735
  End
  Begin TextBox TextSlidingCutchng
    Height          = 285
    HelpContextID   = 1304
    Left            = 3600
    TabIndex        = 19
    Text            = "3.0"
    Top             = 1080
    Width           = 735
  End
  Begin TextBox TextSlidingCutseas
    Height          = 285
    HelpContextID   = 1303
    Left            = 3600
    TabIndex        = 18
    Text            = "3.0"
    Top             = 360
    Width           = 735
  End
  Begin Label Label
    BackStyle       = 0 'Transparent
    Caption         = "For the trading day factors:"
    FontBold        = 0 'False
    FontItalic      = 0 'False
    FontName        = "MS Sans Serif"
    FontSize        = 8.25
    FontStrikethru  = 0 'False
    FontUnderline   = 0 'False
    Height          = 255
    Index           = 4
    Left            = 120
    TabIndex        = 17
    Top             = 1680
    Width           = 1995
  End
  Begin Label Label
    BackStyle       = 0 'Transparent
    Caption         = "For the period-to-period or year-to-year percent changes in seasonally adjusted series:"
    FontBold        = 0 'False
    FontItalic      = 0 'False
    FontName        = "MS Sans Serif"
    FontSize        = 8.25
    FontStrikethru  = 0 'False
    FontUnderline   = 0 'False
    Height          = 435
    Index           = 3
    Left            = 120
    TabIndex        = 16
    Top             = 960
    Width           = 3375
  End
  Begin Label Label
    BackStyle       = 0 'Transparent
    Caption         = "For the seasonal factors and seasonally adjusted series:"
    FontBold        = 0 'False
    FontItalic      = 0 'False
    FontName        = "MS Sans Serif"
    FontSize        = 8.25
    FontStrikethru  = 0 'False
    FontUnderline   = 0 'False
    Height          = 375
    Index           = 2
    Left            = 120
    TabIndex        = 15
    Top             = 360
    Width           = 3075
  End
End
Begin SSCommand Command3DSliding
  BevelWidth      = 1
  Caption         = "&Next"
```

```
    Font3D      = 0 'None
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 375
    HelpContextID = 13
    Index       = 0
    Left        = 7800
    TabIndex    = 0
    Top         = 3240
    Width       = 1095
  End
  Begin TextBox TextSlidingLength
    Height      = 285
    HelpContextID = 1302
    Left        = 4080
    TabIndex    = 5
    Top         = 840
    Width       = 735
  End
  Begin TextBox TextSlidingStart
    Height      = 285
    HelpContextID = 1301
    Left        = 3600
    MaxLength   = 7
    TabIndex    = 3
    Top         = 240
    Width       = 1215
  End
  Begin Label Label
    BackStyle   = 0 'Transparent
    Caption     = "Parameters estimation:"
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    Index       = 7
    Left        = 5040
    TabIndex    = 13
    Top         = 2160
    Width       = 1755
  End
  Begin Label Label
    BackStyle   = 0 'Transparent
    Caption     = "Outlier identification:"
    FontBold    = 0 'False
    FontItalic  = 0 'False
    FontName    = "MS Sans Serif"
    FontSize    = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height      = 195
    Index       = 5
    Left        = 5040
    TabIndex    = 35
    Top         = 1200
    Width       = 1575
  End
  Begin Label Label
    AutoSize    = -1 'True
    BackStyle   = 0 'Transparent
    Caption     = "1/2"
    Height      = 195
    Index       = 8
    Left        = 8700
    TabIndex    = 33
    Top         = 120
```

```
  Width       = 315
End
Begin Label Label
  BackStyle    = 0 'Transparent
  Caption      = " For the dates use the format yyyy.mm (monthly) yyyy.q (quartly)."
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 255
  Index        = 16
  Left         = 840
  TabIndex     = 30
  Top          = 3900
  Width        = 4575
End
Begin Label Label
  BackStyle    = 0 'Transparent
  Caption      = "Note:"
  Height       = 255
  Index        = 15
  Left         = 360
  TabIndex     = 29
  Top          = 3900
  Width        = 495
End
Begin Label Label
  BackStyle    = 0 'Transparent
  Caption      = "Replacement of missing values:"
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 255
  Index        = 6
  Left         = 5040
  TabIndex     = 21
  Top          = 240
  Width        = 2355
End
Begin Label Label
  BackStyle    = 0 'Transparent
  Caption      = "Length of the span of time series data:          (in months or quarters)"
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 495
  Index        = 1
  Left         = 240
  TabIndex     = 4
  Top          = 720
  Width        = 2835
End
Begin Label Label
  BackStyle    = 0 'Transparent
  Caption      = "Start date for sliding spans comparisons:"
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 255
  Index        = 0
  Left         = 240
  TabIndex     = 2
```

```
    Top         = 240
    Width       = 2895
  End
End
Begin SSPanel Panel3DSliding
  BackColor     = &H00C0C0C0&
  Font3D        = 0 'None
  Height        = 4250
  Index         = 1
  Left          = 0
  TabIndex      = 10
  Top           = 0
  Width         = 9255
  Begin SSCommand Command3DSliding
    BevelWidth    = 1
    Caption       = "&Back"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 375
    Index         = 2
    Left          = 7800
    TabIndex      = 31
    Top           = 3720
    Width         = 1095
  End
  Begin SSFrame Frame3DSliding
    Caption       = "Storage:"
    Font3D        = 0 'None
    ForeColor     = &H00000000&
    Height        = 3135
    Index         = 1
    Left          = 360
    TabIndex      = 23
    Top           = 360
    Width         = 5655
    Begin SSCheck Check3DSlidingSave
      Caption       = "Indirect year-to-year changes from all sliding spans"
      Font3D        = 0 'None
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 195
      HelpContextID = 1300
      Index         = 8
      Left          = 180
      TabIndex      = 6
      Top           = 2760
      Width         = 3915
    End
    Begin SSCheck Check3DSlidingSave
      Caption       = "Indirect seasonally adjusted series from all sliding spans"
      Font3D        = 0 'None
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 195
      HelpContextID = 1300
      Index         = 7
      Left          = 180
      TabIndex      = 7
      Top           = 2460
      Width         = 4275
```

```
End
Begin SSCheck Check3DSlidingSave
  Caption      = "Indirect period-to-period changes from all sliding spans"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 1300
  Index        = 6
  Left         = 180
  TabIndex     = 8
  Top          = 2160
  Width        = 4155
End
Begin SSCheck Check3DSlidingSave
  Caption      = "Indirect seasonal factors from all sliding spans"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 1300
  Index        = 5
  Left         = 180
  TabIndex     = 9
  Top          = 1860
  Width        = 3615
End
Begin SSCheck Check3DSlidingSave
  Caption      = "Trading day factors form all sliding spans"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 1300
  Index        = 4
  Left         = 180
  TabIndex     = 28
  Top          = 1560
  Width        = 3195
End
Begin SSCheck Check3DSlidingSave
  Caption      = "Year-to-year changes form all sliding spans"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 1300
  Index        = 3
  Left         = 180
  TabIndex     = 27
  Top          = 1260
  Width        = 3375
End
Begin SSCheck Check3DSlidingSave
  Caption      = "Seasonally adjusted series from all sliding spans"
  Font3D       = 0 'None
  FontBold     = 0 'False
```

```
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    Height         = 195
    HelpContextID  = 1300
    Index          = 2
    Left           = 180
    TabIndex       = 26
    Top            = 960
    Width          = 3855
  End
  Begin SSCheck Check3DSlidingSave
    Caption        = "Period-to-period changes from all sliding spans"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    Height         = 195
    HelpContextID  = 1300
    Index          = 1
    Left           = 180
    TabIndex       = 25
    Top            = 660
    Width          = 3615
  End
  Begin SSCheck Check3DSlidingSave
    Caption        = "Seasonal factors from all sliding spans"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    Height         = 195
    HelpContextID  = 1300
    Index          = 0
    Left           = 180
    TabIndex       = 24
    Top            = 360
    Width          = 2955
  End
End
Begin SSCommand Command3DSliding
  BevelWidth     = 1
  Caption        = "&OK"
  Font3D         = 0  'None
  FontBold       = 0  'False
  FontItalic     = 0  'False
  FontName       = "MS Sans Serif"
  FontSize       = 8.25
  FontStrikethru = 0  'False
  FontUnderline  = 0  'False
  Height         = 375
  Index          = 1
  Left           = 7800
  TabIndex       = 11
  Top            = 3240
  Width          = 1095
End
Begin Label Label
  AutoSize       = -1  True
  BackStyle      = 0  'Transparent
  Caption        = "2/2"
  Height         = 195
  Index          = 19
  Left           = 8700
  TabIndex       = 32
```

```
      Top          =  120
      Width        =  315
    End
  End
End
Option Explicit

Sub cargar_sliding ()
Dim X%

    TextSlidingStart.Text = (getvar("sliding", "start"))
    TextSlidingLength = (getvar("sliding", "length"))
    TextSlidingCutseas = (getvar("sliding", "cutseas"))
    TextSlidingCutchng = (getvar("sliding", "cutchng"))
    TextSlidingCuttd = (getvar("sliding", "cuttd"))
    ComboSlidingFixmissing.ListIndex = Val(getvar("sliding", "fixmissing"))
    'For X% = 0 To 1
    '  Frame3DSliding(X%) = (getvar("sliding", "frame" & X%))
    'Next X%
    ComboSlidingOutlier.ListIndex = Val(getvar("sliding", "outlier"))
    ComboSlidingFixmdl.ListIndex = Val(getvar("sliding", "fixmdl"))
    For X% = 0 To 8
      Check3DSlidingSave(X%) = Val(getvar("sliding", "save" & X%))
    Next X%

End Sub

Sub command3dsliding_click (Index As Integer)
Dim X%, I1%, I2%, I3%

 Select Case Index
   Case 0
    panel3dsliding(1).ZOrder
   Case 1
    'GUARDAR
    checkcargar(14) = "yes"
    Call setvar("sliding", "cargar", checkcargar(14))
    Call setvar("sliding", "start", TextSlidingStart)
    Call setvar("sliding", "length", TextSlidingLength)
    Call setvar("sliding", "cutseas", TextSlidingCutseas)
    Call setvar("sliding", "cutchng", TextSlidingCutchng)
    Call setvar("sliding", "cuttd", TextSlidingCuttd)
    I1% = ComboSlidingFixmissing.ListIndex
    Call setvar("sliding", "fixmissing", I1%)
    For X% = 0 To 1
      Call setvar("sliding", "frame" & X%, Frame3DSliding(X%))
    Next X%
    I2% = ComboSlidingOutlier.ListIndex
    Call setvar("sliding", "outlier", I2%)
    I3% = ComboSlidingFixmdl.ListIndex
    Call setvar("sliding", "fixmdl", I3%)
    For X% = 0 To 8
      Call setvar("sliding", "save" & X%, Check3DSlidingSave(X%))
    Next X%
   Case 2 'back
    panel3dsliding(0).ZOrder
 End Select

End Sub

Sub Form_Load ()

 panel3dsliding(0).ZOrder

 Call combo_sliding

 'If checkcargar(14) = "yes" Then
   Call cargar_sliding
 ' Else
 '   Call inicializar_sliding
 'End If

 Call command3dsliding_click(1)
```

```
End Sub

Sub TextSlidingCutchng_KeyPress (keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextSlidingCutseas_KeyPress (keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextSlidingCuttd_KeyPress (keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextSlidingLength_KeyPress (keyascii As Integer)

   keyascii = validar(keyascii)

End Sub

Sub TextSlidingStart_KeyPress (keyascii As Integer)

keyascii = validar(keyascii)

End Sub

Sub TextSlidingStart_LostFocus ()

   'If Val(TextSlidingStart.Text) < Val(start$) Then
      'mensaje de error
      'textslidingstart.SetFocus mirar!!!
   '   response% = MsgBox("Date not valid", 48, title$)
   'End If

End Sub
```

1.1.17. Xhistory.frm

```
VERSION 2.00
Begin Form history
  BackColor      = &H00C0C0C0&
  BorderStyle    = 0 'None
  ClientHeight   = 4215
  ClientLeft     = 105
  ClientTop      = 1980
  ClientWidth    = 9060
  ControlBox     = 0 'False
  Height         = 4620
  HelpContextID  = 7
  Left           = 45
  LinkTopic      = "Form1"
  MaxButton      = 0 'False
  MinButton      = 0 'False
  ScaleHeight    = 4215
  ScaleWidth     = 9060
  Top            = 1635
  Width          = 9180
  Begin SSPanel Panel3DHistory
    BackColor      = &H00C0C0C0&
    Font3D         = 0 'None
    Height         = 4250
    Index          = 1
    Left           = 0
    TabIndex       = 15
    Top            = 0
    Width          = 9255
    Begin SSCommand Command3DHistory
      BevelWidth     = 1
      Caption        = "&Back"
      Font3D         = 0 'None
      FontBold       = 0 'False
      FontItalic     = 0 'False
      FontName       = "MS Sans Serif"
      FontSize       = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height         = 375
      Index          = 5
      Left           = 7800
      TabIndex       = 70
      Top            = 3720
      Width          = 1095
    End
    Begin SSCheck Check3DHistory
      Caption        = "Restimate the regARIMA model during the revisions history analysis"
      Font3D         = 0 'None
      FontBold       = 0 'False
      FontItalic     = 0 'False
      FontName       = "MS Sans Serif"
      FontSize       = 8.25
      FontStrikethru = 0 'False
      FontUnderline  = 0 'False
      Height         = 255
      HelpContextID  = 707
      Index          = 0
      Left           = 660
      TabIndex       = 29
      Top            = 3480
      Value          = -1 'True
      Width          = 5115
    End
    Begin SSFrame Frame3Dhistory
      Caption        = "Forecasts leads that will be analyzed in the revisions"
      Font3D         = 0 'None
      FontBold       = 0 'False
      FontItalic     = 0 'False
      FontName       = "MS Sans Serif"
      FontSize       = 8.25
      FontStrikethru = 0 'False
```

```
    FontUnderline = 0 'False
    ForeColor     = &H00000000&
    Height        = 1155
    Index         = 1
    Left          = 660
    TabIndex      = 23
    Top           = 2100
    Width         = 4875
    Begin TextBox TextHistoryFstep
      Height        = 315
      HelpContextID = 706
      Index         = 3
      Left          = 3720
      TabIndex      = 28
      Top           = 540
      Width         = 615
    End
    Begin TextBox TextHistoryFstep
      Height        = 315
      HelpContextID = 706
      Index         = 2
      Left          = 2580
      TabIndex      = 27
      Top           = 540
      Width         = 615
    End
    Begin TextBox TextHistoryFstep
      Height        = 315
      HelpContextID = 706
      Index         = 1
      Left          = 1380
      TabIndex      = 26
      Top           = 540
      Width         = 615
    End
    Begin TextBox TextHistoryFstep
      Height        = 315
      HelpContextID = 706
      Index         = 0
      Left          = 180
      TabIndex      = 25
      Top           = 540
      Width         = 615
    End
    Begin Label Label3
      BackStyle     = 0 'Transparent
      Caption       = "analysis of the forecasts:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 255
      Left          = 120
      TabIndex      = 24
      Top           = 240
      Width         = 2295
    End
  End
  Begin TextBox TextHistoryRevisionlag
    Height        = 285
    HelpContextID = 705
    Left          = 3960
    TabIndex      = 21
    Top           = 1620
    Width         = 735
  End
  Begin TextBox TextHistoryEndtable
    Height        = 285
    HelpContextID = 704
    Left          = 3960
    TabIndex      = 19
```

```
  Top         = 1080
  Width       = 1215
End
Begin SSCommand Command3DHistory
  BevelWidth  = 1
  Caption     = "&Next"
  Font3D      = 0 'None
  FontBold    = 0 'False
  FontItalic  = 0 'False
  FontName    = "MS Sans Serif"
  FontSize    = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height      = 375
  Index       = 1
  Left        = 7800
  TabIndex    = 17
  Top         = 3240
  Width       = 1095
End
Begin ComboBox ComboHistoryType
  Height      = 300
  HelpContextID = 703
  Left        = 660
  Style       = 2 'Dropdown List
  TabIndex    = 16
  Top         = 480
  Width       = 4515
End
Begin Label Label
  AutoSize    = -1 'True
  BackStyle   = 0 'Transparent
  Caption     = "2/5"
  Height      = 195
  Index       = 17
  Left        = 8700
  TabIndex    = 78
  Top         = 120
  Width       = 315
End
Begin Label Label
  BackStyle   = 0 'Transparent
  Caption     = " For the dates use the format yyyy.mm (monthly) yyyy.q (quartly)."
  FontBold    = 0 'False
  FontItalic  = 0 'False
  FontName    = "MS Sans Serif"
  FontSize    = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height      = 255
  Index       = 16
  Left        = 1140
  TabIndex    = 66
  Top         = 3900
  Width       = 4575
End
Begin Label Label
  BackStyle   = 0 'Transparent
  Caption     = "Note:"
  Height      = 255
  Index       = 15
  Left        = 660
  TabIndex    = 67
  Top         = 3900
  Width       = 495
End
Begin Label Label2
  BackStyle   = 0 'Transparent
  Caption     = "Position of the observation analyzed relative to the concurrent estimate:"
  FontBold    = 0 'False
  FontItalic  = 0 'False
  FontName    = "MS Sans Serif"
  FontSize    = 8.25
```

```
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height = 435
      Left = 660
      TabIndex = 20
      Top = 1560
      Width = 2715
    End
    Begin Label Label
      BackStyle = 0 'Transparent
      Caption = "Final date of the output tables:"
      FontBold = 0 'False
      FontItalic = 0 'False
      FontName = "MS Sans Serif"
      FontSize = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height = 195
      Index = 2
      Left = 660
      TabIndex = 18
      Top = 1080
      Width = 2355
    End
    Begin Label Label1
      BackStyle = 0 'Transparent
      Caption = "Time points used in the history analysis:"
      FontBold = 0 'False
      FontItalic = 0 'False
      FontName = "MS Sans Serif"
      FontSize = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height = 195
      Left = 600
      TabIndex = 2
      Top = 240
      Width = 2895
    End
  End
  Begin SSPanel Panel3DHistory
    BackColor = &H00C0C0C0&
    Font3D = 0 'None
    Height = 4250
    Index = 0
    Left = 0
    TabIndex = 1
    Top = 0
    Width = 9255
    Begin TextBox TextHistoryStart
      Height = 285
      HelpContextID = 701
      Left = 3960
      TabIndex = 14
      Top = 300
      Width = 1155
    End
    Begin SSCommand Command3DHistory
      BevelWidth = 1
      Caption = "&Next"
      Font3D = 0 'None
      FontBold = 0 'False
      FontItalic = 0 'False
      FontName = "MS Sans Serif"
      FontSize = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height = 375
      HelpContextID = 7
      Index = 0
      Left = 7800
      TabIndex = 0
      Top = 3240
```

```
  Width       = 1095
End
Begin SSFrame Frame3Dhistory
  Caption     = "Estimates analyzed:"
  Font3D      = 0 'None
  ForeColor   = &H00000000&
  Height      = 3075
  Index       = 0
  Left        = 540
  TabIndex    = 3
  Top         = 720
  Width       = 5715
  Begin SSCheck Check3DHistoryEstimates
    Caption       = "702"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    Index         = 7
    Left          = 180
    TabIndex      = 11
    Top           = 2520
    Width         = 195
  End
  Begin SSCheck Check3DHistoryEstimates
    Caption       = "AIC's and maximum log likelihoods for the regARIMA model"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    HelpContextID = 702
    Index         = 6
    Left          = 180
    TabIndex      = 10
    Top           = 2160
    Width         = 4575
  End
  Begin SSCheck Check3DHistoryEstimates
    Caption       = "Period-to-period changes in the final Henderson trend component"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    HelpContextID = 702
    Index         = 5
    Left          = 180
    TabIndex      = 9
    Top           = 1860
    Width         = 4995
  End
  Begin SSCheck Check3DHistoryEstimates
    Caption       = "Final Henderson trend component"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height        = 195
    HelpContextID = 702
```

```
  Index        = 4
  Left         = 180
  TabIndex     = 8
  Top          = 1560
  Width        = 2775
End
Begin SSCheck Check3DHistoryEstimates
  Caption      = "Period-to-period changes in the final seasonally adjusted series"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 702
  Index        = 3
  Left         = 180
  TabIndex     = 7
  Top          = 1260
  Width        = 4755
End
Begin SSCheck Check3DHistoryEstimates
  Caption      = "Final seasonal factors, projected one year ahead"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 702
  Index        = 2
  Left         = 180
  TabIndex     = 6
  Top          = 960
  Width        = 3795
End
Begin SSCheck Check3DHistoryEstimates
  Caption      = "Final seasonal factors"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 702
  Index        = 1
  Left         = 180
  TabIndex     = 5
  Top          = 660
  Width        = 1875
End
Begin SSCheck Check3DHistoryEstimates
  Caption      = "Final seasonally adjusted series"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 702
  Index        = 0
  Left         = 180
  TabIndex     = 4
  Top          = 360
  Value        = -1 'True
```

```
      Width        = 2595
    End
    Begin Label Label
      BackStyle    = 0 'Transparent
      Caption      = "Forecasts and evolving mean square forecast errors generated from the regARIMA model"
      FontBold     = 0 'False
      FontItalic   = 0 'False
      FontName     = "MS Sans Serif"
      FontSize     = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height       = 435
      Index        = 0
      Left         = 420
      TabIndex     = 12
      Top          = 2460
      Width        = 4575
    End
  End
  Begin Label Label
    AutoSize     = -1 'True
    BackStyle    = 0 'Transparent
    Caption      = "1/5"
    Height       = 195
    Index        = 12
    Left         = 8700
    TabIndex     = 75
    Top          = 120
    Width        = 315
  End
  Begin Label Label
    BackStyle    = 0 'Transparent
    Caption      = " For the dates use the format yyyy.mm (monthly) yyyy.q (quartly)."
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height       = 255
    Index        = 10
    Left         = 1020
    TabIndex     = 68
    Top          = 3900
    Width        = 4575
  End
  Begin Label Label
    BackStyle    = 0 'Transparent
    Caption      = "Note:"
    Height       = 255
    Index        = 9
    Left         = 540
    TabIndex     = 69
    Top          = 3900
    Width        = 495
  End
  Begin Label Label
    BackStyle    = 0 'Transparent
    Caption      = "Starting date of the revisions history analysis:"
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height       = 255
    Index        = 1
    Left         = 540
    TabIndex     = 13
    Top          = 300
    Width        = 3315
  End
End
```

```
Begin SSPanel Panel3DHistory
  BackColor      = &H00C0C0C0&
  Font3D         = 0 'None
  Height         = 4250
  Index          = 4
  Left           = 0
  TabIndex       = 53
  Top            = 0
  Width          = 9255
  Begin SSCommand Command3DHistory
    BevelWidth   = 1
    Caption      = "&Back"
    Font3D       = 0 'None
    FontBold     = 0 'False
    FontItalic   = 0 'False
    FontName     = "MS Sans Serif"
    FontSize     = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height       = 375
    Index        = 8
    Left         = 7800
    TabIndex     = 73
    Top          = 3720
    Width        = 1095
  End
  Begin SSFrame Frame3Dhistory
    Caption      = "Storage (continued):"
    Font3D       = 0 'None
    ForeColor    = &H00000000&
    Height       = 3375
    Index        = 3
    Left         = 300
    TabIndex     = 55
    Top          = 180
    Width        = 7215
    Begin SSCheck Check3DHistorySave
      Caption    = "Concurrent and most recent estimate of the projected seasonal factors"
      Font3D     = 0 'None
      FontBold   = 0 'False
      FontItalic = 0 'False
      FontName   = "MS Sans Serif"
      FontSize   = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height     = 195
      HelpContextID = 700
      Index      = 15
      Left       = 180
      TabIndex   = 22
      Top        = 2940
      Width      = 5415
    End
    Begin SSCheck Check3DHistorySave
      Caption    = "Concurrent and most recent estimate of the period-to-period changes in the trend component"
      Font3D     = 0 'None
      FontBold   = 0 'False
      FontItalic = 0 'False
      FontName   = "MS Sans Serif"
      FontSize   = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height     = 195
      HelpContextID = 700
      Index      = 14
      Left       = 180
      TabIndex   = 61
      Top        = 2520
      Width      = 6795
    End
    Begin SSCheck Check3DHistorySave
      Caption    = "Concurrent and most recent estimate of the trend component"
      Font3D     = 0 'None
```

```
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 700
    Index         = 13
    Left          = 180
    TabIndex      = 60
    Top           = 2100
    Width         = 4695
  End
  Begin SSCheck Check3DHistorySave
    Caption       = "Concurrent and most recent estimate of the period-to-period"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 700
    Index         = 12
    Left          = 180
    TabIndex      = 59
    Top           = 1680
    Width         = 4575
  End
  Begin SSCheck Check3DHistorySave
    Caption       = "Concurrent and most recent estimate of the seasonal factors"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 700
    Index         = 11
    Left          = 180
    TabIndex      = 58
    Top           = 1260
    Width         = 4635
  End
  Begin SSCheck Check3DHistorySave
    Caption       = "Concurrent and most recent estimate of the seasonally adjusted data"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    Height        = 195
    HelpContextID = 700
    Index         = 10
    Left          = 180
    TabIndex      = 57
    Top           = 840
    Width         = 5415
  End
  Begin SSCheck Check3DHistorySave
    Caption       = "Record of seasonal filter selection for each observation in the revisions history"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
```

```
        FontUnderline  =  0  'False
        Height         =  195
        HelpContextID  =  700
        Index          =  9
        Left           =  180
        TabIndex       =  56
        Top            =  420
        Width          =  5955
      End
    End
    Begin SSCommand Command3DHistory
      BevelWidth     =  1
      Caption        =  "&OK"
      Font3D         =  0  'None
      FontBold       =  0  'False
      FontItalic     =  0  'False
      FontName       =  "MS Sans Serif"
      FontSize       =  8.25
      FontStrikethru =  0  'False
      FontUnderline  =  0  'False
      Height         =  375
      Index          =  4
      Left           =  7800
      TabIndex       =  54
      Top            =  3240
      Width          =  1095
    End
    Begin Label Label
      AutoSize       =  -1  'True
      BackStyle      =  0  'Transparent
      Caption        =  "5/5"
      Height         =  195
      Index          =  11
      Left           =  8700
      TabIndex       =  74
      Top            =  120
      Width          =  315
    End
  End
  Begin SSPanel Panel3DHistory
    BackColor      =  &H00C0C0C0&
    Font3D         =  0  'None
    Height         =  4250
    Index          =  3
    Left           =  0
    TabIndex       =  38
    Top            =  0
    Width          =  9255
    Begin SSCommand Command3DHistory
      BevelWidth     =  1
      Caption        =  "&Back"
      Font3D         =  0  'None
      FontBold       =  0  'False
      FontItalic     =  0  'False
      FontName       =  "MS Sans Serif"
      FontSize       =  8.25
      FontStrikethru =  0  'False
      FontUnderline  =  0  'False
      Height         =  375
      Index          =  7
      Left           =  7800
      TabIndex       =  72
      Top            =  3720
      Width          =  1095
    End
    Begin SSFrame Frame3Dhistory
      Caption        =  "Storage:"
      Font3D         =  0  'None
      ForeColor      =  &H00000000&
      Height         =  3975
      Index          =  2
      Left           =  420
      TabIndex       =  40
```

```
 Top         = 60
 Width       = 6795
 Begin SSCheck Check3DHistorySave
   Caption      = "Revision from concurrent to most recent estimate of the projected seasonal factors"
   Font3D       = 0 'None
   FontBold     = 0 'False
   FontItalic   = 0 'False
   FontName     = "MS Sans Serif"
   FontSize     = 8.25
   FontStrikethru = 0 'False
   FontUnderline = 0 'False
   Height       = 195
   HelpContextID = 700
   Index        = 8
   Left         = 240
   TabIndex     = 49
   Top          = 3600
   Value        = -1 'True
   Width        = 6255
 End
 Begin SSCheck Check3DHistorySave
   Font3D       = 0 'None
   FontBold     = 0 'False
   FontItalic   = 0 'False
   FontName     = "MS Sans Serif"
   FontSize     = 8.25
   FontStrikethru = 0 'False
   FontUnderline = 0 'False
   Height       = 195
   HelpContextID = 700
   Index        = 7
   Left         = 240
   TabIndex     = 48
   Top          = 3180
   Value        = -1 'True
   Width        = 195
 End
 Begin SSCheck Check3DHistorySave
   Caption      = "Evolving mean forecast errors for each observation in the revisions history"
   Font3D       = 0 'None
   FontBold     = 0 'False
   FontItalic   = 0 'False
   FontName     = "MS Sans Serif"
   FontSize     = 8.25
   FontStrikethru = 0 'False
   FontUnderline = 0 'False
   Height       = 195
   HelpContextID = 700
   Index        = 6
   Left         = 240
   TabIndex     = 47
   Top          = 2820
   Value        = -1 'True
   Width        = 5595
 End
 Begin SSCheck Check3DHistorySave
   Caption      = "Log-LIkelihood and AIC Values for each observation in the revisions history"
   Font3D       = 0 'None
   FontBold     = 0 'False
   FontItalic   = 0 'False
   FontName     = "MS Sans Serif"
   FontSize     = 8.25
   FontStrikethru = 0 'False
   FontUnderline = 0 'False
   Height       = 195
   HelpContextID = 700
   Index        = 5
   Left         = 240
   TabIndex     = 46
   Top          = 2460
   Value        = -1 'True
   Width        = 5715
 End
```

```
Begin SSCheck Check3DHistorySave
  Caption      = "Revision from concurrent  to most recent estimate of the trend-component"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 700
  Index        = 4
  Left         = 240
  TabIndex     = 45
  Top          = 2100
  Value        = -1 'True
  Width        = 5595
End
Begin SSCheck Check3DHistorySave
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 700
  Index        = 3
  Left         = 240
  TabIndex     = 44
  Top          = 1560
  Value        = -1 'True
  Width        = 255
End
Begin SSCheck Check3DHistorySave
  Caption      = "Revision from concurrent to most recent estimate of the seasonal factors"
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 700
  Index        = 2
  Left         = 240
  TabIndex     = 43
  Top          = 1200
  Value        = -1 'True
  Width        = 5415
End
Begin SSCheck Check3DHistorySave
  Font3D       = 0 'None
  FontBold     = 0 'False
  FontItalic   = 0 'False
  FontName     = "MS Sans Serif"
  FontSize     = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  Height       = 195
  HelpContextID = 700
  Index        = 1
  Left         = 240
  TabIndex     = 42
  Top          = 720
  Value        = -1 'True
  Width        = 255
End
Begin SSCheck Check3DHistorySave
  Caption      = "Record for outliers removed and kept for the revisions history"
  Font3D       = 0 'None
```

```
      FontBold      = 0  'False
      FontItalic    = 0  'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0  'False
      FontUnderline = 0  'False
      Height        = 195
      HelpContextID = 700
      Index         = 0
      Left          = 240
      TabIndex      = 41
      Top           = 360
      Value         = -1  'True
      Width         = 4755
    End
    Begin Label Label
      BackStyle     = 0  'Transparent
      Caption       = "Revision from concurrent to most recent estimate of the period-to-period changes in the trend component"
      FontBold      = 0  'False
      FontItalic    = 0  'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0  'False
      FontUnderline = 0  'False
      Height        = 435
      Index         = 6
      Left          = 540
      TabIndex      = 52
      Top           = 3180
      Width         = 5355
    End
    Begin Label Label
      BackStyle     = 0  'Transparent
      Caption       = "Revision from concurrent to most recent estimate of the period-to-period changes in the seasonally
adjusted data"
      FontBold      = 0  'False
      FontItalic    = 0  'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0  'False
      FontUnderline = 0  'False
      Height        = 435
      Index         = 5
      Left          = 540
      TabIndex      = 51
      Top           = 1560
      Width         = 5235
    End
    Begin Label Label
      BackStyle     = 0  'Transparent
      Caption       = "Revision from concurrent to most recent estimate of the seasonally adjusted data"
      FontBold      = 0  'False
      FontItalic    = 0  'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0  'False
      FontUnderline = 0  'False
      Height        = 435
      Index         = 4
      Left          = 540
      TabIndex      = 50
      Top           = 720
      Width         = 4935
    End
  End
  Begin SSCommand Command3DHistory
    BevelWidth    = 1
    Caption       = "&Next"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
```

```
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height = 375
    Index = 3
    Left = 7800
    TabIndex = 39
    Top = 3240
    Width = 1095
  End
  Begin Label Label
    AutoSize = -1 'True
    BackStyle = 0 'Transparent
    Caption = "4/5"
    Height = 195
    Index = 13
    Left = 8700
    TabIndex = 76
    Top = 120
    Width = 315
  End
End
Begin SSPanel Panel3DHistory
  BackColor = &H00C0C0C0&
  Font3D = 0 'None
  Height = 4250
  Index = 2
  Left = 0
  TabIndex = 30
  Top = 0
  Width = 9255
  Begin SSCommand Command3DHistory
    BevelWidth = 1
    Caption = "&Back"
    Font3D = 0 'None
    FontBold = 0 'False
    FontItalic = 0 'False
    FontName = "MS Sans Serif"
    FontSize = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    Height = 375
    Index = 6
    Left = 7800
    TabIndex = 71
    Top = 3720
    Width = 1095
  End
  Begin ComboBox ComboHistoryAdjfcst
    Height = 300
    HelpContextID = 709
    Left = 600
    Style = 2 'Dropdown List
    TabIndex = 65
    Top = 1200
    Width = 5295
  End
  Begin ComboBox ComboHistoryRefresh
    Height = 300
    HelpContextID = 708
    Left = 600
    Style = 2 'Dropdown List
    TabIndex = 63
    Top = 420
    Width = 5295
  End
  Begin SSCheck Check3DHistory
    Caption = "Values of missing observations will be recalculated each time model estimation"
    Font3D = 0 'None
    FontBold = 0 'False
    FontItalic = 0 'False
    FontName = "MS Sans Serif"
    FontSize = 8.25
    FontStrikethru = 0 'False
```

```
  FontUnderline  =  0  'False
  Height         =  195
  HelpContextID  =  713
  Index          =  2
  Left           =  600
  TabIndex       =  37
  Top            =  3720
  Width          =  5895
End
Begin SSCheck Check3DHistory
  Caption        =  "Remove all level shift and ramp outlier effects"
  Font3D         =  0  'None
  FontBold       =  0  'False
  FontItalic     =  0  'False
  FontName       =  "MS Sans Serif"
  FontSize       =  8.25
  FontStrikethru =  0  'False
  FontUnderline  =  0  'False
  Height         =  195
  HelpContextID  =  712
  Index          =  1
  Left           =  600
  TabIndex       =  36
  Top            =  3240
  Value          =  -1  'True
  Width          =  3735
End
Begin TextBox TextHistoryOutlierwin
  Height         =  285
  HelpContextID  =  711
  Left           =  4800
  TabIndex       =  35
  Top            =  2700
  Width          =  675
End
Begin ComboBox ComboHistoryOutlier
  Height         =  300
  HelpContextID  =  710
  Left           =  600
  Style          =  2  'Dropdown List
  TabIndex       =  32
  Top            =  2040
  Width          =  5295
End
Begin SSCommand Command3DHistory
  BevelWidth     =  1
  Caption        =  "&Next"
  Font3D         =  0  'None
  FontBold       =  0  'False
  FontItalic     =  0  'False
  FontName       =  "MS Sans Serif"
  FontSize       =  8.25
  FontStrikethru =  0  'False
  FontUnderline  =  0  'False
  Height         =  375
  Index          =  2
  Left           =  7800
  TabIndex       =  31
  Top            =  3240
  Width          =  1095
End
Begin Label Label
  AutoSize       =  -1  'True
  BackStyle      =  0  'Transparent
  Caption        =  "3/5"
  Height         =  195
  Index          =  14
  Left           =  8700
  TabIndex       =  77
  Top            =  120
  Width          =  315
End
Begin Label Label
```

```
      BackStyle     = 0 'Transparent
      Caption       = "Prior adjustments and forecasts:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 255
      Index         = 8
      Left          = 600
      TabIndex      = 64
      Top           = 960
      Width         = 2355
    End
    Begin Label Label
      BackStyle     = 0 'Transparent
      Caption       = "Initial values used for parameter estimation:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 255
      Index         = 7
      Left          = 600
      TabIndex      = 62
      Top           = 180
      Width         = 3195
    End
    Begin Label Label
      BackStyle     = 0 'Transparent
      Caption       = "Number of observations used for outlier identification:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 255
      Index         = 3
      Left          = 600
      TabIndex      = 34
      Top           = 2700
      Width         = 3735
    End
    Begin Label Label4
      BackStyle     = 0 'Transparent
      Caption       = "Outlier detection in the revisions history analysis:"
      FontBold      = 0 'False
      FontItalic    = 0 'False
      FontName      = "MS Sans Serif"
      FontSize      = 8.25
      FontStrikethru = 0 'False
      FontUnderline = 0 'False
      Height        = 255
      Left          = 600
      TabIndex      = 33
      Top           = 1800
      Width         = 3495
    End
  End
End
Option Explicit

Sub cargar_history ()
Dim X%

  texthistorystart.Text = (getvar("history", "start"))
  For X% = 0 To 7
    check3dhistoryestimates(X%) = Val(getvar("history", "checkestimates" & X%))
  Next X%
```

```
    'For X% = 0 To 2
    '   frame3dhistory(X%) = (getvar("history", "frameestimates" & X%))
    'Next X%
    combohistorytype.ListIndex = Val(getvar("history", "combotype"))
    texthistoryendtable = (getvar("history", "endtable"))
    texthistoryrevisionlag.Text = (getvar("history", "revisionlag"))
    For X% = 0 To 3
       texthistoryfstep(X%) = (getvar("history", "fstep" & X%))
    Next X%
    For X% = 0 To 2
       check3dhistory(X%) = Val(getvar("history", "checkhistory" & X%))
    Next X%
    For X% = 0 To 15
       check3dhistorysave(X%) = Val(getvar("history", "save" & X%))
    Next X%
    combohistoryoutlier.ListIndex = Val(getvar("history", "outlier"))
    combohistoryrefresh.ListIndex = Val(getvar("history", "refresh"))
    combohistoryadjfcst.ListIndex = Val(getvar("history", "adjfcst"))
    texthistoryoutlierwin.Text = (getvar("history", "outlierwin"))

End Sub

Sub command3dhistory_click (Index As Integer)
Dim X%, I1%, I2%, I3%, I4%

    Select Case Index
       Case 0
          panel3dhistory(1).ZOrder
       Case 1
          panel3dhistory(2).ZOrder
       Case 2
          panel3dhistory(3).ZOrder
       Case 3
          panel3dhistory(4).ZOrder
       Case 4
          'GUARDAR
          checkcargar(15) = "yes"
          Call setvar("history", "cargar", checkcargar(15))
          Call setvar("history", "start", texthistorystart)
          For X% = 0 To 7
             Call setvar("history", "checkestimates" & X%, check3dhistoryestimates(X%))
          Next X%
          For X% = 0 To 2
             Call setvar("history", "frameestimates" & X%, frame3dhistory(X%))
          Next X%
          I1% = combohistorytype.ListIndex
          Call setvar("history", "combotype", I1%)
          Call setvar("history", "endtable", texthistoryendtable)
          Call setvar("history", "revisionlag", texthistoryrevisionlag)
          For X% = 0 To 3
             Call setvar("history", "fstep" & X%, texthistoryfstep(X%))
          Next X%
          For X% = 0 To 2
             Call setvar("history", "checkhistory" & X%, check3dhistory(X%))
          Next X%
          For X% = 0 To 15
             Call setvar("history", "save" & X%, check3dhistorysave(X%))
          Next X%
          I2% = combohistoryoutlier.ListIndex
          Call setvar("history", "outlier", I2%)
          I3% = combohistoryrefresh.ListIndex
          Call setvar("history", "refresh", I3%)
          I4% = combohistoryadjfcst.ListIndex
          Call setvar("history", "adjfcst", I4%)
          Call setvar("history", "outlierwin", texthistoryoutlierwin)
       Case 5 'back segundo panel
          panel3dhistory(0).ZOrder
       Case 6 'back tercer panel
          panel3dhistory(1).ZOrder
       Case 7 'back cuarto panel
          panel3dhistory(2).ZOrder
       Case 8 'back quinto panel
          panel3dhistory(3).ZOrder
```

```
    End Select
End Sub

Sub Form_Load ()

    panel3dhistory(0).ZOrder

    Call combo_history

    'If checkcargar(15) = "yes" Then
       Call cargar_history
    '   Else Call inicializar_history
    'End If

    Call command3dhistory_click(4)

End Sub

Sub Label_Click (Index As Integer) 'n

  Select Case Index
    Case 0
      check3dhistoryestimates(7).Value = Not (check3dhistoryestimates(7).Value)
    Case 4
      check3dhistorysave(1).Value = Not (check3dhistorysave(1).Value)
    Case 5
      check3dhistorysave(3).Value = Not (check3dhistorysave(3).Value)
    Case 6
      check3dhistorysave(7).Value = Not (check3dhistorysave(7).Value)
  End Select

End Sub

Sub TextHistoryEndtable_KeyPress (keyascii As Integer)

    keyascii = validar(keyascii)

End Sub

Sub TextHistoryFstep_KeyPress (Index As Integer, keyascii As Integer)

    keyascii = validar(keyascii)

End Sub

Sub TextHistoryOutlierwin_KeyPress (keyascii As Integer)

    keyascii = validar(keyascii)

End Sub

Sub TextHistoryRevisionlag_KeyPress (keyascii As Integer)

    keyascii = validar(keyascii)

End Sub

Sub TextHistoryStart_KeyPress (keyascii As Integer)

    keyascii = validar(keyascii)

End Sub
```

1.1.18. Tables.frm

```
VERSION 2.00
Begin Form Tables
  BackColor      = &H00C0C0C0&
  ClientHeight   = 5445
  ClientLeft     = 90
  ClientTop      = 1395
  ClientWidth    = 9450
  Height         = 5850
  Left           = 30
  LinkTopic      = "Form1"
  ScaleHeight    = 5445
  ScaleWidth     = 9450
  Top            = 1050
  Width          = 9570
  Begin SSPanel Panel3DTables
    BackColor      = &H00C0C0C0&
    Font3D         = 2 'Raised w/heavy shading
    ForeColor      = &H00000000&
    Height         = 5475
    Index          = 0
    Left           = 0
    TabIndex       = 0
    Top            = 0
    Width          = 9435
    Begin ListBox ListSeries
      Height         = 1200
      Left           = 480
      TabIndex       = 28
      Top            = 840
      Width          = 2175
    End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Original series"
    Font3D         = 0 'None
    FontBold       = 0 'False
    FontItalic     = 0 'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0 'False
    FontUnderline  = 0 'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 0
    Left           = 2880
    TabIndex       = 27
    Top            = 240
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Transformed series"
    Font3D         = 0 'None
    FontBold       = 0 'False
    FontItalic     = 0 'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0 'False
    FontUnderline  = 0 'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 1
    Left           = 2880
    TabIndex       = 26
    Top            = 600
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Seasonally adjusted series"
    Font3D         = 0 'None
```

```
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 3
    Left          = 2880
    TabIndex      = 25
    Top           = 1320
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Trend-cycle"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 4
    Left          = 2880
    TabIndex      = 24
    Top           = 1680
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Irregular component"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 5
    Left          = 4920
    TabIndex      = 23
    Top           = 240
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Seasonal factors"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 6
    Left          = 4920
    TabIndex      = 22
    Top           = 600
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Trading day"
    Font3D        = 0  'None
    FontBold      = 0  'False
```

```
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 7
    Left           = 4920
    TabIndex       = 21
    Top            = 960
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Si-ratios (differences)"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 9
    Left           = 4920
    TabIndex       = 20
    Top            = 1680
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Outliers"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 10
    Left           = 6960
    TabIndex       = 19
    Top            = 240
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "Forecasts"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
    FontStrikethru = 0  'False
    FontUnderline  = 0  'False
    ForeColor      = &H00000000&
    Height         = 375
    Index          = 11
    Left           = 6960
    TabIndex       = 18
    Top            = 600
    Width          = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth     = 1
    Caption        = "ACF/PACF, Residuals"
    Font3D         = 0  'None
    FontBold       = 0  'False
    FontItalic     = 0  'False
```

```
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 2
    Left          = 2880
    TabIndex      = 17
    Top           = 960
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Holiday factors"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 8
    Left          = 4920
    TabIndex      = 16
    Top           = 1320
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Slidingspans"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 12
    Left          = 6960
    TabIndex      = 15
    Top           = 960
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "History"
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0  'False
    FontUnderline = 0  'False
    ForeColor     = &H00000000&
    Height        = 375
    Index         = 13
    Left          = 6960
    TabIndex      = 14
    Top           = 1320
    Width         = 2055
  End
  Begin SSCommand Command3DGraph
    BevelWidth    = 1
    Caption       = "Composite"
    Enabled       = 0  'False
    Font3D        = 0  'None
    FontBold      = 0  'False
    FontItalic    = 0  'False
```

```
  FontName      = "MS Sans Serif"
  FontSize      = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  ForeColor     = &H00000000&
  Height        = 375
  Index         = 14
  Left          = 6960
  TabIndex      = 13
  Top           = 1680
  Width         = 2055
End
Begin SSCommand Command3DClearList
  BevelWidth    = 1
  Caption       = "&Clear list"
  Font3D        = 0 'None
  FontBold      = 0 'False
  FontItalic    = 0 'False
  FontName      = "MS Sans Serif"
  FontSize      = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  ForeColor     = &H00000000&
  Height        = 375
  Left          = 540
  TabIndex      = 11
  Top           = 4920
  Width         = 1095
End
Begin ListBox ListGraph
  Height        = 2175
  Left          = 180
  MultiSelect   = 1 'Simple
  TabIndex      = 10
  Top           = 2400
  Width         = 6075
End
Begin SSCommand Command3DReturn
  BevelWidth    = 1
  Caption       = "&Exit"
  Font3D        = 0 'None
  FontBold      = 0 'False
  FontItalic    = 0 'False
  FontName      = "MS Sans Serif"
  FontSize      = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  ForeColor     = &H00000000&
  Height        = 375
  Left          = 8040
  TabIndex      = 8
  Top           = 4920
  Width         = 1095
End
Begin CommonDialog CMDialog1
  Filter        = "Todos los archivos (*.*)|*.*"
  Left          = 8760
  Top           = 0
End
Begin SSCommand Command3DSameGraph
  BevelWidth    = 1
  Caption       = "&Graph"
  Font3D        = 0 'None
  FontBold      = 0 'False
  FontItalic    = 0 'False
  FontName      = "MS Sans Serif"
  FontSize      = 8.25
  FontStrikethru = 0 'False
  FontUnderline = 0 'False
  ForeColor     = &H00000000&
  Height        = 375
  Left          = 2400
  TabIndex      = 7
```

```
    Top          = 4920
    Width        = 1095
  End
  Begin Label Label
    BackStyle    = 0  'Transparent
    Caption      = "Series:"
    Height       = 255
    Index        = 0
    Left         = 480
    TabIndex     = 12
    Top          = 180
    Width        = 615
  End
  Begin Label LabelSeries
    BorderStyle  = 1  'Fixed Single
    Height       = 285
    Left         = 480
    TabIndex     = 29
    Top          = 420
    Width        = 2175
  End
  Begin Label Label
    AutoSize     = -1  'True
    BackStyle    = 0  'Transparent
    Caption      = "Graphics:"
    Height       = 195
    Index        = 1
    Left         = 180
    TabIndex     = 9
    Top          = 2160
    Width        = 825
  End
End
Begin SSPanel Panel3DTables
  Font3D       = 2  'Raised w/heavy shading
  ForeColor    = &H00000000&
  Height       = 5475
  Index        = 6
  Left         = 0
  TabIndex     = 6
  Top          = 0
  Width        = 9435
End
Begin SSPanel Panel3DTables
  Font3D       = 2  'Raised w/heavy shading
  ForeColor    = &H00000000&
  Height       = 5475
  Index        = 5
  Left         = 0
  TabIndex     = 5
  Top          = 0
  Width        = 9435
End
Begin SSPanel Panel3DTables
  Font3D       = 2  'Raised w/heavy shading
  ForeColor    = &H00000000&
  Height       = 5475
  Index        = 4
  Left         = 0
  TabIndex     = 4
  Top          = 0
  Width        = 9435
End
Begin SSPanel Panel3DTables
  Font3D       = 2  'Raised w/heavy shading
  ForeColor    = &H00000000&
  Height       = 5475
  Index        = 3
  Left         = 0
  TabIndex     = 3
  Top          = 0
  Width        = 9435
End
```

```
  Begin SSPanel Panel3DTables
    Font3D       = 2 'Raised w/heavy shading
    ForeColor    = &H00000000&
    Height       = 5475
    Index        = 2
    Left         = 0
    TabIndex     = 2
    Top          = 0
    Width        = 9435
  End
  Begin SSPanel Panel3DTables
    Font3D       = 2 'Raised w/heavy shading
    ForeColor    = &H00000000&
    Height       = 5475
    Index        = 1
    Left         = 0
    TabIndex     = 1
    Top          = 0
    Width        = 9435
  End
End
Option Explicit

Sub Command3DClearList_Click ()
Dim x%

  listgraph.Clear

End Sub

Sub Command3DGraph_Click (Index As Integer)

 mousepointer = 11
  Select Case Index
        Case 0 'original series
         If Val(getvar("formx12", "arguments0")) <> 0 Then
         If Val(getvar("series", "save0")) <> 0 Then
          If Dir$(app.Path & "\output\" & seriesgraph & ".A1") <> "" Then
           listgraph.AddItem "Time series"
          End If
         End If
         End If
         If Val(getvar("formx12", "arguments0")) <> 0 Then
         If Val(getvar("series", "save1")) <> 0 Then
          If Dir$(app.Path & "\output\" & seriesgraph & ".MVA") <> "" Then
           listgraph.AddItem "Time series with missing values replaced"
          End If
         End If
         End If
         If Val(getvar("formx12", "arguments3")) <> 0 Then
         If Val(getvar("x11", "checkx112")) <> 0 Then
          If Dir$(app.Path & "\output\" & seriesgraph & ".B1") <> "" Then
           listgraph.AddItem "Original series, adjusted for prior effects and forecast extended"
          End If

         End If
         End If

         If Val(getvar("formx12", "arguments3")) <> 0 Then
         If Val(getvar("x11", "checkx1117")) <> 0 Then
          If Dir$(app.Path & "\output\" & seriesgraph & ".E4") <> "" Then
           listgraph.AddItem "Ratio of yearly totals for original, seasonally adjusted series"
          End If

         End If
         End If
         If Val(getvar("formx12", "arguments3")) <> 0 Then
         If Val(getvar("x11", "checkx1118")) <> 0 Then
          If Dir$(app.Path & "\output\" & seriesgraph & ".E5") <> "" Then
           listgraph.AddItem "Percent changes in original series"
          End If

         End If
```

```
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1135")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B19") <> "" Then
   listgraph.AddItem "Original series adjusted for preliminary trading day"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1137")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C1") <> "" Then
   listgraph.AddItem "Original series modified for outliers, td and prior factors, C iteration"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1147")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C19") <> "" Then
   listgraph.AddItem "Original series adjusted for final trading day"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1149")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".D1") <> "" Then
   listgraph.AddItem "Original series modified for outliers, td and prior factors, D iteration"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1157")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E1") <> "" Then
   listgraph.AddItem "Original series modified for extreme values"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments12")) <> 0 Then
 If Val(getvar("regadjust", "save5")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A11") <> "" Then
   listgraph.AddItem "Original series adjusted for regARIMA outliers"
  End If
 End If
 End If
Case 1 'transformed series
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save0")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A2") <> "" Then
   listgraph.AddItem "Prior adjustment factors"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save1")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A3") <> "" Then
   listgraph.AddItem "Prior adjusted series"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save2")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".TRN") <> "" Then
   listgraph.AddItem "Prior adjusted and transformed series"
   ' datagraph$ = app.path & "\output\" & seriesgraph & ".TRN"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save3")) <> 0 Then
```

```
    If Dir$(app.Path & "\output\" & seriesgraph & ".A10") <> "" Then
     listgraph.AddItem "Henderson trend of prior adjusted series"
     'datagraph$  = app.path & "\output\" & seriesgraph & ".A10"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments12")) <> 0 Then
   If Val(getvar("regadjust", "save4")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".USR") <> "" Then
     listgraph.AddItem "Factors from user-defined regression variables"
    End If
   End If
   End If
  Case 2 'ACF/PACF, residuals
   If Val(getvar("formx12", "arguments10")) <> 0 Then
   If Val(getvar("check", "save0")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".ACF") <> "" Then
     listgraph.AddItem "ACF"
     'datagraph$  = app.path & "\output\" & seriesgraph & ".ACF"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments10")) <> 0 Then
   If Val(getvar("check", "save1")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".PCF") <> "" Then
     listgraph.AddItem "PACF"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".PCF"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments8")) <> 0 Then
   If Val(getvar("estimate", "checkestimate0")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".RSD") <> "" Then
     listgraph.AddItem "Residuals"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".RSD"
    End If
   End If
   End If
  Case 3 'Seasonally adjusted series
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1110")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D11") <> "" Then
     listgraph.AddItem "Final seasonally adjusted series"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D11"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1111")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".SAA") <> "" Then
     listgraph.AddItem "Final seasonally adjusted series with constrained yearly totals"
     'datagraph$= app.path & "\output\" & seriesgraph & ".SAA"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1112")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".RND") <> "" Then
     listgraph.AddItem "Rounded final seasonally adjusted series"
     'datagraph$= app.path & "\output\" & seriesgraph & ".RND"
    End If

   End If
   End If

   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1116")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".FAD") <> "" Then
     listgraph.AddItem "Final adjustment difference"
     'datagraph$= app.path & "\output\" & seriesgraph & ".RND"
    End If

   End If
```

```
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1117")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E4") <> "" Then
   listgraph.AddItem "Ratio of yearly totals for original, seasonally adjusted series"
   'datagraph$= app.path & "\output\" & seriesgraph & ".RND"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1119")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6") <> "" Then
   listgraph.AddItem "Percent changes in seasonally adjusted series"
   'datagraph$= app.path & "\output\" & seriesgraph & ".E6"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1120")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6A") <> "" Then
   listgraph.AddItem "Percent changes in seasonally adjusted series with revised yearly totals"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".E6A"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1121")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6R") <> "" Then
   listgraph.AddItem "Percent changes in rounded seasonally adjusted series"
   'datagraph$  = app.path & "\output\" & seriesgraph & ".E6R"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1126")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B6") <> "" Then
   listgraph.AddItem "Preliminary seasonally adjusted series, B iteration"
   'datagraph$  = app.path & "\output\" & seriesgraph & ".B6"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1130")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B11") <> "" Then
   listgraph.AddItem "Seasonally adjusted series, B iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".B11"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1141")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C6") <> "" Then
   listgraph.AddItem "Preliminary seasonally adjusted series, C iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".C6"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1145")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C11") <> "" Then
   listgraph.AddItem "Seasonally adjusted series, C iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".C11"
  End If

 End If
 End If
```

```
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1153")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".D6") <> "" Then
   listgraph.AddItem "Preliminary seasonally adjusted series, D iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".D6"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1158")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E2") <> "" Then
   listgraph.AddItem "Seasonally adjusted series modified for extreme values"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".E2"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1160")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E11") <> "" Then
   listgraph.AddItem "Robust final seasonally adjusted series"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".E11"
  End If

 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1161")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".F1") <> "" Then
   listgraph.AddItem "MCD moving average of the final seasonally adjusted series"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".F1"
  End If
 End If
 End If
Case 4 'trend cycle
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1113")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".D12") <> "" Then
   listgraph.AddItem "Final trend-cycle"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".D12"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1122")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E7") <> "" Then
   listgraph.AddItem "Final trend-cycle using original series modified for extreme values"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".E7"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1123")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B2") <> "" Then
   listgraph.AddItem "Preliminary trend-cycle, B iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".B2"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1127")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B7") <> "" Then
   listgraph.AddItem "Preliminary trend-cycle, B iteration ??"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".B7"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1138")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C2") <> "" Then
   listgraph.AddItem "Preliminary trend-cycle, C iteration"
   'datagraph$ = app.path & "\output\" & seriesgraph & ".C2"
```

```
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1142")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C7") <> "" Then
     listgraph.AddItem "Preliminary trend-cycle, C iteration ??"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C7"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1150")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D2") <> "" Then
     listgraph.AddItem "Preliminary trend-cycle, D iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D2"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1154")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D7") <> "" Then
     listgraph.AddItem "Preliminary trend-cycle, D iteration ??"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D7"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1155")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".TAL") <> "" Then
     listgraph.AddItem "Final trend-cycle adjusted for LS"
     'datagraph$ = app.path & "\output\" & seriesgraph & "TAL"
    End If
   End If
   End If
  Case 5 'irregular component
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx114")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C17") <> "" Then
     listgraph.AddItem "Final weights for the irregular component"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C17"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1114")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D13") <> "" Then
     listgraph.AddItem "Final irregular component"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D13"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1131")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B13") <> "" Then
     listgraph.AddItem "Irregular component, B iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B13"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1133")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B17") <> "" Then
     listgraph.AddItem "Preliminary weights for the irregular component"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B17"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1146")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C13") <> "" Then
     listgraph.AddItem "Irregular component, C iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C13"
```

```
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1156")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".IAO") <> "" Then
     listgraph.AddItem "Final irregular component adjusted for AO"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".IAO"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1159")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".E3") <> "" Then
     listgraph.AddItem "Irregular component modified for extreme values"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".E3"
    End If
   End If
   End If
  Case 6 'seasonal component
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx118")) <> 0 = True Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D10") <> "" Then
     listgraph.AddItem "Final seasonal factors"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D10"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx119")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".FSD") <> "" Then
     listgraph.AddItem "Final seasonal difference"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".FSD"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1115")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D16") <> "" Then
     listgraph.AddItem "Combined seasonal and trading day factors"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D16"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1125")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B5") <> "" Then
     listgraph.AddItem "Preliminary seasonal factors, B iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B5"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1129")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B10") <> "" Then
     listgraph.AddItem "Seasonal factors, B iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B10"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1140")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C5") <> "" Then
     listgraph.AddItem "Preliminary seasonal factors, C iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C5"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1152")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D5") <> "" Then
     listgraph.AddItem "Preliminary seasonal factors, D iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D5"
```

```
    End If
   End If
   End If
  Case 7 'trading day
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx111")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".A4") <> "" Then
     listgraph.AddItem "Prior trading day factors and weights"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".A4"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx113")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C16") <> "" Then
     listgraph.AddItem "Final trading day factors and weights"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C16"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx115")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C18") <> "" Then
     listgraph.AddItem "Final trading day from combined daily weights"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C18"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1115")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D16") <> "" Then
     listgraph.AddItem "Combined seasonal and trading day factors"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D16"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1132")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B16") <> "" Then
     listgraph.AddItem "Preliminary trading day factors and weights"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B16"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1134")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B18") <> "" Then
     listgraph.AddItem "Preliminary trading day from combined daily weights"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B18"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1135")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B19") <> "" Then
     listgraph.AddItem "Original series adjusted for preliminary trading day"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B19"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments12")) <> 0 Then
   If Val(getvar("regadjust", "save2")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".TD") <> "" Then
     listgraph.AddItem "regARIMA Trading day factors"
    End If
   End If
   End If
  Case 8 'holiday factors
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx110")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".H1") <> "" Then
     listgraph.AddItem "Combined X-11 holiday adjustment factors"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".H1"
```

```
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments12")) <> 0 Then
   If Val(getvar("regadjust", "save3")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".HOL") <> "" Then
     listgraph.AddItem "regARIMA Holiday factors"
    End If
   End If
   End If
  Case 9 'si-ratios (differences)
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx116")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D8") <> "" Then
     listgraph.AddItem "Final unmodified si-ratios"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D8"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx117")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D9") <> "" Then
     listgraph.AddItem "Final replacement values for si-ratios, D iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D9"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1124")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B3") <> "" Then
     listgraph.AddItem "Preliminary unmodified si-ratios"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B3"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1128")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".B8") <> "" Then
     listgraph.AddItem "Unmodified si-ratios"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".B8"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1139")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C4") <> "" Then
     listgraph.AddItem "Modified si-ratios, C iteration ??"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C4"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1143")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".C9") <> "" Then
     listgraph.AddItem "Modified si-ratios, C iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".C9"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments3")) <> 0 Then
   If Val(getvar("x11", "checkx1151")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".D4") <> "" Then
     listgraph.AddItem "Modified si-ratios, D iteration"
     'datagraph$ = app.path & "\output\" & seriesgraph & ".D4"
    End If
   End If
   End If
  Case 10 'extreme value, outliers
   If Val(getvar("formx12", "arguments12")) <> 0 Then
   If Val(getvar("regadjust", "save0")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".AO") <> "" Then
     listgraph.AddItem "regARIMA Additive outlier factors"
    End If
```

```
    End If
    End If
    If Val(getvar("formx12", "arguments12")) <> 0 Then
    If Val(getvar("regadjust", "save1")) <> 0 Then
     If Dir$(app.Path & "\output\" & seriesgraph & ".LS") <> "" Then
      listgraph.AddItem "regARIMA Level change and ramp factors"
     End If
    End If
    End If
   Case 11 'forecasts
    If Val(getvar("formx12", "arguments11")) <> 0 Then
    If Val(getvar("forecast", "save0")) <> 0 Then
     If Dir$(app.Path & "\output\" & seriesgraph & ".FTR") <> "" Then
      listgraph.AddItem "Forecasts on the transformated scale"
     End If
    End If
    End If
   If Val(getvar("formx12", "arguments11")) <> 0 Then
   If Val(getvar("forecast", "save1")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".FVR") <> "" Then
     listgraph.AddItem "Forecast error variances on the transformated scale"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments11")) <> 0 Then
   If Val(getvar("forecast", "save2")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph & ".FCT") <> "" Then
     listgraph.AddItem "Forecasts on the original scale"
    End If
   End If
   End If
  Case 12 'slidingspans
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save0")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFS") <> "" Then
     listgraph.AddItem "Seasonal factors (sliding spans)"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save1")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHS") <> "" Then
          listgraph.AddItem "Period-to-period changes (sliding spans)"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save2")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAS") <> "" Then
          listgraph.AddItem "Seasonally adjusted series (sliding spans)"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save3")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".YCS") <> "" Then
          listgraph.AddItem "Year-to-year changes (sliding spans)"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save4")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".TDS") <> "" Then
          listgraph.AddItem "Trading day factors (sliding spans)"
    End If
   End If
   End If
   If Val(getvar("formx12", "arguments13")) <> 0 Then
   If Val(getvar("sliding", "save5")) <> 0 Then
    If Dir$(app.Path & "\output\" & seriesgraph$ & ".SIS") <> "" Then
          listgraph.AddItem "Indirect seasonal factors (sliding spans)"
    End If
   End If
```

```
 End If
 If Val(getvar("formx12", "arguments13")) <> 0 Then
 If Val(getvar("sliding", "save6")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CIS") <> "" Then
          listgraph.AddItem "Indirect period-to-period changes (sliding spans)"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments13")) <> 0 Then
 If Val(getvar("sliding", "save7")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".AIS") <> "" Then
          listgraph.AddItem "Indirect seasonally adjusted series (sliding spans)"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments13")) <> 0 Then
 If Val(getvar("sliding", "save8")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".YIS") <> "" Then
          listgraph.AddItem "Indirect year-to-year changes (sliding spans)"
  End If
 End If
 End If
Case 13 'history
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save0")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".ROT") <> "" Then
   listgraph.AddItem "Record for outliers"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save1")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFH") <> "" Then
          listgraph.AddItem "Revision of seasonally adjusted data"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save2")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAR") <> "" Then
          listgraph.AddItem "Revision of seasonal factors"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save3")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAE") <> "" Then
          listgraph.AddItem "Revision of period-to-period changes in seas.adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save4")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFR") <> "" Then
          listgraph.AddItem "Revision of trend-component"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save5")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFE") <> "" Then
          listgraph.AddItem "Log-likelihood and AIC for observ."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save6")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHR") <> "" Then
          listgraph.AddItem "Mean forecast errors for observ."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
```

```
 If Val(getvar("history", "save7")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHE") <> "" Then
          listgraph.AddItem "Revision of period-to-period changes in trend"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save8")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TRR") <> "" Then
          listgraph.AddItem "Revision of projected seasonal factors"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save9")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TRE") <> "" Then
          listgraph.AddItem "Record of seasonal filter selection"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save10")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".LKH") <> "" Then
          listgraph.AddItem "Estimates of seasonally adjusted data"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save11")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".FCE") <> "" Then
          listgraph.AddItem "Estimates of seasonal factors"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save12")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TCR") <> "" Then
          listgraph.AddItem "Estimates of period-to-period"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save13")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TCE") <> "" Then
          listgraph.AddItem "Estimates of trend component"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save14")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".PSR") <> "" Then
          listgraph.AddItem "Estimates of period-to-period changes in trend"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments14")) <> 0 Then
 If Val(getvar("history", "save15")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".PSE") <> "" Then
          listgraph.AddItem "Estimates of the projected seasonal factors"
  End If
 End If
 End If
Case 14 'composite
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save0")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CMS") <> "" Then
          listgraph.AddItem "Aggregated time series data"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save1")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".ID8") <> "" Then
```

```
            listgraph.AddItem "Final unmodified si-ratios for the ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save2")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".ISF") <> "" Then
            listgraph.AddItem "Final seasonal factors for the ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save3")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".ISA") <> "" Then
            listgraph.AddItem "Final indirect seasonally adjusted series"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save4")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IAA") <> "" Then
            listgraph.AddItem "Final ind. seas. adj. series, with yearly totals adjusted"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save5")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IRN") <> "" Then
            listgraph.AddItem "Rounded indirect fianl seasonally adjusted series"
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save6")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".ITN") <> "" Then
            listgraph.AddItem "Final trend-cycle for the ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save7")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IIR") <> "" Then
            listgraph.AddItem "Final irregular component for the ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save8")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE1") <> "" Then
            listgraph.AddItem "Original series modified for extreme values from the ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save9")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE2") <> "" Then
            listgraph.AddItem "Seas. adj. series modified for extreme values from ind. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save10")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE3") <> "" Then
            listgraph.AddItem "Irregular component modified for extreme values from the ind. seas. adj."
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments15")) <> 0 Then
 If Val(getvar("composite", "save11")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE4") <> "" Then
            listgraph.AddItem "Percent changes in the original series"
  End If
 End If
```

```
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save12")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE5") <> "" Then
                    listgraph.AddItem "Ratio of yearly totals for original, ind. adj. series"
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save13")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE6") <> "" Then
                    listgraph.AddItem "Percent changes in the ind. seas. adj. series"
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save14")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".I6A") <> "" Then
                    listgraph.AddItem "Percent changes for ind. seas. adj. series with revised yearly totals"
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save15")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".I6R") <> "" Then
                    listgraph.AddItem "Percent changes for rounded ind. seas. adj. series"
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save16")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".IE7") <> "" Then
                    listgraph.AddItem "Trend-cycle computed using series modified for extremes in ind. adj."
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save17")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".IEE") <> "" Then
                    listgraph.AddItem "Robust final ind. seas. adj. series"
             End If
            End If
            End If
            If Val(getvar("formx12", "arguments15")) <> 0 Then
            If Val(getvar("composite", "save18")) <> 0 Then
             If Dir$(app.Path & "\output\" & seriesgraph$ & ".IF1") <> "" Then
                    listgraph.AddItem "MCD moving average of the final ind. seas. adj. series"
             End If
            End If
            End If
  End Select
 mousepointer = 0

End Sub

Sub Command3DReturn_Click ()

 tables.Hide

End Sub

Sub Command3DSameGraph_Click ()
Dim x%, gNum%

 mousepointer = 11 'n
 Unload plot
 Unload graphvalues
 col% = 0
 ReDim grapharray(25, 600)
 listgraph.ListIndex = 0
 Erase GraphArrayLen
 gNum% = 0
 For x% = 0 To listgraph.ListCount - 1
```

```
    If listgraph.Selected(x%) = True Then
      'Unload plot
      ReDim Preserve GraphArrayLen(gNum%)
      GraphArrayLen(gNum%) = cargar_datos()
      gNum% = gNum% + 1
      'plot.Show
    End If
    If listgraph.ListIndex < listgraph.ListCount - 1 Then
      listgraph.ListIndex = listgraph.ListIndex + 1
    End If
  Next x%
  plot.Show
  plot.Panel3DNameSeries.Caption = "Time Series " & seriesgraph$
  mousepointer = 0 'n

End Sub

Sub Form_Activate ()
Dim x%

'   'cargar las series
'   ListSeries.Clear
'   If menuprincipal.ListSeries.List(0) <> "" Then
'     ListSeries.AddItem series_name(CStr(menuprincipal.ListSeries.List(0)))
'   End If
'   If menuprincipal.ListSeriesSingleSpec.List(0) <> "" Then
'     For X% = 0 To menuprincipal.ListSeriesSingleSpec.ListCount - 1
'       ListSeries.AddItem series_name(CStr(menuprincipal.ListSeriesSingleSpec.List(X%)))
'     Next X%
'   End If
'   'poner alguna serie por defecto en el
'   LabelSeries.caption = ListSeries.List(0)
'
'   'falta el de multi spec mode
'
'   'cargar los datos de gráficos posibles

    'cargar las series
    ListSeries.Clear
    For x% = 0 To menuprincipal.List.ListCount - 1
      If menuprincipal.List.List(x%) <> "" Then
            ListSeries.AddItem series_name(CStr(menuprincipal.List.List(x%)))
      End If
    Next x%
    'If menuprincipal.ListSeriesSingleSpec.List(0) <> "" Then
    ' For X% = 0 To menuprincipal.ListSeriesSingleSpec.ListCount - 1
    '   ListSeries.AddItem series_name(CStr(menuprincipal.ListSeriesSingleSpec.List(X%)))
    ' Next X%
    'End If
    'poner alguna serie por defecto en el LabelSeries
    LabelSeries.Caption = ListSeries.List(0)

    'falta el de multi spec mode

    'cargar los datos de gráficos posibles
    Call list_graph

End Sub

Sub Form_Load ()
Dim fichero$, x%

  Call titleform
  Me.Caption = titletables$

'   col% = 0
'   ReDim grapharray(25, 400)
'
'   'cargar las series
'   ListSeries.Clear
'   If menuprincipal.ListSeries.List(0) <> "" Then
'     ListSeries.AddItem series_name(CStr(menuprincipal.ListSeries.List(0)))
'   End If
```

```
'   If menuprincipal.ListSeriesSingleSpec.List(0) <> "" Then
'     For X% = 0 To menuprincipal.ListSeriesSingleSpec.ListCount - 1
'       ListSeries.AddItem series_name(CStr(menuprincipal.ListSeriesSingleSpec.List(X%)))
'     Next X%
'   End If
'   'poner alguna serie por defecto en el LabelSeries
'   LabelSeries.caption = ListSeries.List(0)
'
    'falta el de multi spec mode
'
'   'cargar los datos de gráficos posibles
'
    col% = 0
    ReDim grapharray(25, 400)

    'cargar las series
    ListSeries.Clear
    For x% = 0 To menuprincipal.List.ListCount - 1
      If menuprincipal.List.List(x%) <> "" Then
            ListSeries.AddItem series_name(CStr(menuprincipal.List.List(x%)))
      End If
    Next x%
    'If menuprincipal.ListSeriesSingleSpec.List(0) <> "" Then
    '  For X% = 0 To menuprincipal.ListSeriesSingleSpec.ListCount - 1
    '    ListSeries.AddItem series_name(CStr(menuprincipal.ListSeriesSingleSpec.List(X%)))
    '  Next X%
    'End If
    'poner alguna serie por defecto en el LabelSeries
    LabelSeries.Caption = ListSeries.List(0)

    'falta el de multi spec mode

    'cargar los datos de gráficos posibles
    Call list_graph
    Me.Left = 15
    Me.Top = 1050

End Sub

Sub LabelSeries_Change ()

  seriesgraph$ = LabelSeries.Caption

End Sub

Sub ListSeries_Click ()

    LabelSeries.Caption = ListSeries.List(ListSeries.ListIndex)
    seriesgraph$ = LabelSeries.Caption
    Call list_graph

End Sub
```

1.1.19. Plot.frm

```
VERSION 2.00
Begin Form Plot
  BackColor     = &H00C0C0C0&
  Caption       = "Plot"
  ClientHeight  = 5070
  ClientLeft    = 75
  ClientTop     = 1770
  ClientWidth   = 9360
  Height        = 5475
  Left          = 15
  LinkTopic     = "Form1"
  ScaleHeight   = 5070
  ScaleWidth    = 9360
  Top           = 1425
  Width         = 9480
  Begin SSPanel Panel3DNameSeries
    BevelOuter    = 1 'Inset
    ForeColor     = &H00000000&
    Height        = 315
    Left          = 2400
    TabIndex      = 3
    Top           = 60
    Width         = 2835
  End
  Begin SSCommand Command3DPlot
    BevelWidth    = 1
    Caption       = "&Exit"
    ForeColor     = &H00000000&
    Height        = 375
    Left          = 7680
    RoundedCorners = 0 'False
    TabIndex      = 2
    Top           = 1440
    Width         = 1095
  End
  Begin SSCommand Command3DValues
    BevelWidth    = 1
    Caption       = "Values"
    ForeColor     = &H00000000&
    Height        = 375
    Left          = 7680
    RoundedCorners = 0 'False
    TabIndex      = 1
    Top           = 900
    Width         = 1095
  End
  Begin VtChart VtChart1
    AutoIncrement = 0 'False
    Column        = 1
    ColumnLabelIndex= 1
    ErrorOffset   = 20000
    FileName      = PLOT.FRX:0000
    Height        = 4575
    Left          = 60
    RandomFill    = -1 'True
    Repaint       = -1 'True
    Row           = 1
    RowLabelIndex = 1
    TabIndex      = 0
    Top           = 420
    Width         = 7455
  End
End
Option Explicit

Sub Command3DPlot_Click ()

  Unload graphvalues
  tables.Show
  Unload Me
```

```
End Sub

Sub Command3DValues_Click ()

  graphvalues.Visible = True
  graphvalues.ZOrder

End Sub

Sub ChartData ()
Dim Range%, MaxVal#, MinVal#
Dim X%, Y%, errorCode%, ssError%', GraphData#()
'Dim clpstr$
Dim prevpos%, devpos%
Dim prevVal#, devVal#, RealValue#
Dim Period%
Dim OrderCount%, Count%, IsRealData%, IsForecastData%
Dim Buffer$, Buffsize%
Dim LastRow%, LastCol%
Dim Pen As VtcPen
Dim Color As VtcColor
Dim PointBrush As VtcBrush
Dim PointPen As VtcPen
Dim PointMarker As VtcMarker
Dim ForeDate As Variant
Dim rect As VtcRect

  'Nº de series:
  errorCode% = VtChSetColumnCount(vtchart1.Handle, Col%)

  'Set Labels for the chart
  vtchart1.ColumnLabelCount = 1
  vtchart1.RowLabelCount = 2


  ssError = SSClearRange(graphvalues.SheetGraphData.SS, -1, -I, -1, -1, 1)

  MaxVal# = GraphArray(0, 0)
  MinVal# = MaxVal#

  'Ponemos los datos de las series:
  For X% = 0 To UBound(GraphArray, 2)

     For Y% = 0 To UBound(GraphArrayLen)
        If X% <= GraphArrayLen(Y%) - 2 Then
          ssError = SSSetNumberRC(graphvalues.SheetGraphData.SS, X% + 2, Y% + 3, GraphArray(Y%, X))
          If MaxVal# < GraphArray(Y%, X) Then MaxVal# = GraphArray(Y%, X)
          If MinVal# > GraphArray(Y%, X) Then MinVal# = GraphArray(Y%, X)
        End If
     Next Y%

     'ssError = SSSetTextRC(GraphValues.SheetGraphData.SS, x% + 2, 1, " " & Right$(Format$(Period%), 2))
     'ssError = SSSetTextRC(GraphValues.SheetGraphData.SS, x% + 2, 2, " " & Right$(Format$(OrderCount%), 2))

     "Period and Order count
     'OrderCount% = OrderCount% + 1
     'If OrderCount% <> 0 And OrderCount% > SeriesIds(0).Periodicity Then
     '  Period% = Period% + 1
     '  OrderCount% = 1
     'End If

  Next X%

  MaxVal# = MaxVal# + (MaxVal# - MinVal#) * .05
  MinVal# = MinVal# - (MaxVal# - MinVal#) * .05

  errorCode% = VtChSetAxisValueScale(vtchart1.Handle, VtChAxisIdY, 1, False, MinVal#, MaxVal#, 3, 0)
  'errorCode% = VtChSetAxisCategoryScale(VtChart1.Handle, VtChAxisIdX, 1, False, SeriesIds(0).Periodicity,
SeriesIds(0).Periodicity)

  ' Get the last row and last column of the worksheet
  ' and create a Formula One reference
  ssError = SSGetLastRow(graphvalues.SheetGraphData.SS, LastRow)
```

```
 ssError = SSGetLastCol(graphvalues.SheetGraphData.SS, LastCol)

 errorCode% = VtChSetRowCount(vtchart1.Handle, LastRow)

 'Series colors:
 Call SetSeriesColors(UBound(GraphArray) + 1)
 For Y% = 0 To Col%
  errorCode% = VtChGetSeriesPen(vtchart1.Handle, Y% + 1, Pen)
  Pen.Width = 1
  errorCode% = VtChSetSeriesPen(vtchart1.Handle, Y% + 1, Pen)
 Next Y%


 'Linkatge de la fulla de càlcul amb el gràfic:
 Buffer$ = Space$(10)
 Buffsize% = 10

 ssError = SSFormatRCNr(graphvalues.SheetGraphData.SS, LastRow, LastCol, False, Buffer$, Buffsize%)
 Buffer$ = "A1:" + Buffer$

 ' Tell First Impression the worksheet, range, and parse method
 vtchart1.SsLinkSheet = graphvalues.SheetGraphData.TableName
 vtchart1.SsLinkRange = Buffer$
 vtchart1.SsLinkMode = VtChSsLinkModeOn

 'ssError% = VtChSetLegendLocation(VtChart1.Handle, True, VtChLocationTypeTopRight, Rect)

End Sub

Sub Form_Load ()

 Call titleform
 Me.Caption = titleplot$
 Call ChartData
 Me.Left = 15
 Me.Top = 1050

End Sub

Sub SetSeriesColors (NumSeries%)
Dim Pen As VtcPen
Dim errorCode%

 'Pen.Color.Red = 0
 'Pen.Color.Green = 0
 'Pen.Color.Blue = 255
 'Pen.Style = VtPenStyleSolid
 'Pen.Width = 0
 'ErrorCode% = VtChSetSeriesPen(VtChart1.Handle, 1, Pen)
 If NumSeries% > -1 Then
    Pen.Color.Red = 0
    Pen.Color.Green = 0
    Pen.Color.Blue = 255
    Pen.Style = VtPenStyleSolid
    errorCode% = VtChSetSeriesPen(vtchart1.Handle, 1, Pen)
 End If
 If NumSeries% > 0 Then
    Pen.Color.Red = 0'128
    Pen.Color.Green = 255'0
    Pen.Color.Blue = 0'128
    Pen.Style = VtPenStyleSolid
    errorCode% = VtChSetSeriesPen(vtchart1.Handle, 2, Pen)
 End If
 If NumSeries% > 1 Then
    Pen.Color.Red = 255
    Pen.Color.Green = 0
    Pen.Color.Blue = 255
    Pen.Style = VtPenStyleSolid
    errorCode% = VtChSetSeriesPen(vtchart1.Handle, 3, Pen)
 End If
 If NumSeries% > 2 Then
    Pen.Color.Red = 255
    Pen.Color.Green = 0
```

```
    Pen.Color.Blue = 0
    Pen.Style = VtPenStyleSolid
    errorCode% = VtChSetSeriesPen(vtchart1.Handle, 4, Pen)
  End If

End Sub
```

1.1.20. Graphval.frm

```
VERSION 2.00
Begin Form GraphValues
  BackColor      = &H00808080&
  Caption        = "Values"
  ClientHeight   = 5055
  ClientLeft     = 210
  ClientTop      = 1785
  ClientWidth    = 9330
  Height         = 5460
  Left           = 150
  LinkTopic      = "Form1"
  ScaleHeight    = 5055
  ScaleWidth     = 9330
  Top            = 1440
  Width          = 9450
  Begin SSCommand Command3DGraphval
    BevelWidth     = 1
    Caption        = "&Exit"
    Font3D         = 0 'None
    ForeColor      = &H00000000&
    Height         = 375
    Left           = 1200
    RoundedCorners = 0 'False
    TabIndex       = 6
    Top            = 4620
    Width          = 1095
  End
  Begin SSView SheetGraphData
    AllowAppLaunch = -1 'True
    DataAutoAddNew = -1 'True
    DataConnected  = -1 'True
    DataFields     = ""
    DataHdrField   = ""
    DataRowsBuffered= 64
    DataSetColumnFormats= -1 'True
    DataSetColumnNames= -1 'True
    DataSetColumnWidths= -1 'True
    DataSetMaxCol  = -1 'True
    DataSetMaxRow  = -1 'True
    DoCancelEdit   = -1 'True
    DoClick        = -1 'True
    DoDataNewRow   = -1 'True
    DoDataRowLoad  = -1 'True
    DoDblClick     = -1 'True
    DoEndEdit      = -1 'True
    DoEndRecalc    = -1 'True
    DoObjClick     = -1 'True
    DoObjDblClick  = -1 'True
    DoRClick       = -1 'True
    DoRDblClick    = -1 'True
    DoSelChange    = -1 'True
    DoStartEdit    = -1 'True
    DoStartRecalc  = -1 'True
    DoTopLeftChanged= -1 'True
    EditName       = "SSEdit1"
    FileName       = GRAPHVAL.FRX:0000
    Height         = 4515
    Left           = 0
    TabIndex       = 0
    TableName      = "Dades"
    Top            = 0
    Width          = 7695
  End
  Begin SSCommand Command3DCopyPaste
    BevelWidth     = 1
    Caption        = "Imprimir"
    Font3D         = 0 'None
    FontBold       = 0 'False
    FontItalic     = 0 'False
    FontName       = "MS Sans Serif"
    FontSize       = 8.25
```

```
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    ForeColor     = &H00000000&
    Height        = 315
    Index         = 3
    Left          = 3420
    RoundedCorners = 0 'False
    TabIndex      = 4
    Top           = 4620
    Width         = 1515
  End
  Begin CommonDialog CMDialog1
    Left          = 8640
    Top           = 4500
  End
  Begin SSCommand Command3DCopyPaste
    BevelWidth    = 1
    Caption       = "Exportar a Excel"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    ForeColor     = &H00000000&
    Height        = 315
    Index         = 2
    Left          = 5040
    RoundedCorners = 0 'False
    TabIndex      = 3
    Top           = 4620
    Width         = 1455
  End
  Begin SSCommand Command3DCopyPaste
    BevelWidth    = 1
    Caption       = "Pegar"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    ForeColor     = &H00000000&
    Height        = 315
    Index         = 0
    Left          = 600
    RoundedCorners = 0 'False
    TabIndex      = 2
    Top           = 4620
    Width         = 855
  End
  Begin SSCommand Command3DCopyPaste
    BevelWidth    = 1
    Caption       = "Copiar"
    Font3D        = 0 'None
    FontBold      = 0 'False
    FontItalic    = 0 'False
    FontName      = "MS Sans Serif"
    FontSize      = 8.25
    FontStrikethru = 0 'False
    FontUnderline = 0 'False
    ForeColor     = &H00000000&
    Height        = 315
    Index         = 1
    Left          = 1740
    RoundedCorners = 0 'False
    TabIndex      = 1
    Top           = 4620
    Width         = 855
  End
  Begin Label UM
```

```
    BorderStyle    =  1  'Fixed Single
    Height         =  195
    Left           =  0
    TabIndex       =  5
    Top            =  4680
    Visible        =  0  'False
    Width          =  615
  End
End
Option Explicit

Sub Command3DCopyPaste_Click (Index As Integer)
Dim ssError%
Dim ssError1%

 On Error GoTo Command3DCopyPasteErr

 Select Case Index%
  Case 0
   ssError% = SSEditPaste(SheetGraphData.SS)
  Case 1
   ssError% = SSEditCopy(SheetGraphData.SS)
  Case 2
   CmDialog1.Flags = &H4&
   CmDialog1.Filename = "*.XLS"
   'CmDialog1.DialogTitle = Language$("excel_export") & " 4"
   CmDialog1.DefaultExt = ".XLS"
   CmDialog1.Action = 2
   If Len(CmDialog1.Filename) <> 0 And UCase(CmDialog1.Filename) <> "*.XLS" Then
    SheetGraphData.WriteExcel4 = CmDialog1.Filename
    MsgBox "Fulla exportada a " & CmDialog1.Filename, 64
   End If
  Case 3
   ssError1% = SSFilePrint(SheetGraphData.SS, False)
 End Select

Command3DCopyPasteErr:

End Sub

Sub Command3DGraphval_Click ()

 graphvalues.Hide

End Sub

Sub Form_Load ()

 Call titleform
 Me.Caption = titlegraphval$
 Me.Left = 15
 Me.Top = 1050

End Sub

Sub Form_QueryUnload (Cancel As Integer, UnloadMode As Integer)

 If UnloadMode = 0 Then
  Cancel = True
  Me.Hide
  'Me.WindowState = 1
 End If

End Sub

Sub Form_Resize ()

 On Error Resume Next

 Command3DCopyPaste(2).Left = Me.Width - Command3DCopyPaste(2).Width - 200
 Command3DCopyPaste(3).Left = Command3DCopyPaste(2).Left - Command3DCopyPaste(3).Width - 100
 Command3DCopyPaste(1).Left = Command3DCopyPaste(3).Left - Command3DCopyPaste(1).Width - 100
 Command3DCopyPaste(0).Left = Command3DCopyPaste(1).Left - Command3DCopyPaste(0).Width - 100
```

```
    Command3DCopyPaste(2).Top = Me.Height - Command3DCopyPaste(0).Height - 500
    Command3DCopyPaste(3).Top = Me.Height - Command3DCopyPaste(0).Height - 500
    Command3DCopyPaste(1).Top = Me.Height - Command3DCopyPaste(0).Height - 500
    Command3DCopyPaste(0).Top = Me.Height - Command3DCopyPaste(0).Height - 500

    SheetGraphData.Width = Me.Width - 300

    SheetGraphData.Height = Command3DCopyPaste(2).Top - SheetGraphData.Top - 200

End Sub
```

1.2. FICHEROS .BAS (Módulos)

1.2.1. XGeneral.bas

```
Option Explicit
Global periodicidad$, start$, title$, response%, maxmod%, maxao%, maxls%, maxrp%, maxuser%
Global checkcargar(15)
Global Actualform As Form
Global titlemenu$, titleformx12$, titlegraph$, titleseries$, titlecomposite$
Global titletransform$, titlex11$, titleidentify$, titleregression$, titlearima$
Global titleautomdl$, titleestimate$, titleoutlier$, titlecheck$, titleforecast$
Global titleregadjust$, titleslidingspans$, titlehistory$
Global titleerror$, titleoutput$, titlegraphval$, titleplot$, titletables$
'Global grapharray!(), garpharraylen%(), datagraph$, seriesgraph$, col%'ponerlo en otro .bas

Declare Function SetParent Lib "User" (ByVal hwndchild As Integer, ByVal hwndnewparent As Integer) As Integer
Declare Function OsGetPrivateProfileString% Lib "Kernel" Alias "GetPrivateProfileString" (ByVal AppName$, ByVal KeyName$,
ByVal keydefault$, ByVal ReturnString$, ByVal NumBytes%, ByVal FileName$)
Declare Function OSWritePrivateProfileString% Lib "Kernel" Alias "WritePrivateProfileString" (ByVal AppName$, ByVal
KeyName$, ByVal keydefault$, ByVal FileName$)
Declare Function getmoduleusage% Lib "Kernel" (ByVal hModule%)
Global X12IniFile$
Global CRLF$, Tabul$

Sub combo_arima ()

  arima.ComboArimaFix.Clear
  arima.ComboArimaFix.AddItem "regression and ARMA parameters"
  arima.ComboArimaFix.AddItem "ARMA parameters"
  arima.ComboArimaFix.AddItem "none of the parameters"

End Sub

Sub combo_automdl ()

  automdl.ComboAutomdl(0).Clear
  automdl.ComboAutomdl(0).AddItem "produce a year of forecasts"
  automdl.ComboAutomdl(0).AddItem "produce a year of forecasts and backcasts"
  automdl.ComboAutomdl(1).Clear
  automdl.ComboAutomdl(1).AddItem "first model that satisfies the criteria"
  automdl.ComboAutomdl(1).AddItem "model with the lowest within-sample forecast error"
  automdl.ComboAutomdl(2).Clear
  automdl.ComboAutomdl(2).AddItem "outlier identification is done for each model"
  automdl.ComboAutomdl(2).AddItem "outlier identification is done only in the first model"
  automdl.ComboAutomdl(3).Clear
  automdl.ComboAutomdl(3).AddItem "out-of-sample forecasts errors"
  automdl.ComboAutomdl(3).AddItem "within-sample forecasts errors"

End Sub

Sub combo_composite ()

  composite.ComboCompositeSpectrumtype.Clear
  composite.ComboCompositeSpectrumtype.AddItem "periodogram"
  composite.ComboCompositeSpectrumtype.AddItem "AR-spectrum"

End Sub

Sub combo_estimate ()

  estimate.ComboEstimateParms.Clear
  estimate.ComboEstimateParms.AddItem "estimate model parameters"
  estimate.ComboEstimateParms.AddItem "evaluate the likelihood function"
  estimate.ComboEstimateExact.Clear
  estimate.ComboEstimateExact.AddItem "exact for AR and MA parameters"
  estimate.ComboEstimateExact.AddItem "exact for MA but conditional for AR"
  estimate.ComboEstimateExact.AddItem "conditional for AR and MA parameters"
  estimate.ComboEstimateOutofsample.Clear
  estimate.ComboEstimateOutofsample.AddItem "out-of-sample forecasts errors"
  estimate.ComboEstimateOutofsample.AddItem "within-sample forecasts errors"
```

```
End Sub

Sub combo_history ()

  history.ComboHistoryType.Clear
  history.ComboHistoryType.AddItem "points corresponding to the calendar month or quarter"
  history.ComboHistoryType.AddItem "all points"
  history.ComboHistoryRefresh.Clear
  history.ComboHistoryRefresh.AddItem "parameters estimation from the last model evaluation"
  history.ComboHistoryRefresh.AddItem "estimates derived from the entire series"
  history.ComboHistoryAdjfcst.Clear
  history.ComboHistoryAdjfcst.AddItem "prior adjustments are incorporated into the forecasts"
  history.ComboHistoryAdjfcst.AddItem "the forecasts are produced for the prior adjusted series"
  history.ComboHistoryOutlier.Clear
  history.ComboHistoryOutlier.AddItem "all outliers keep in the RegARIMA model" 'n
  history.ComboHistoryOutlier.AddItem "all outliers removed and not estimated" 'n
  history.ComboHistoryOutlier.AddItem "outliers out of the period for outlier identification removed" 'n

End Sub

Sub combo_outlier ()

  outlier.ComboOutlierTypes.Clear
  outlier.ComboOutlierTypes.AddItem "ao"
  outlier.ComboOutlierTypes.AddItem "ls"
  outlier.ComboOutlierTypes.AddItem "all"
  outlier.ComboOutlierMethod.Clear
  outlier.ComboOutlierMethod.AddItem "addone"
  outlier.ComboOutlierMethod.AddItem "addall"

End Sub

Sub combo_regadjust ()

  regadjust.ComboRegadjustAugmentusertd.Clear
  regadjust.ComboRegadjustAugmentusertd.AddItem "a combination of all trading day factors"
  regadjust.ComboRegadjustAugmentusertd.AddItem "trading day factors derived from the user-defined variables"

End Sub

Sub combo_regression ()

  regression.ComboRegressionFormat.Clear
  regression.ComboRegressionFormat.AddItem "free format"
  regression.ComboRegressionFormat.AddItem "fortran format"
  regression.ComboRegressionFormat.AddItem "edit-date"
  regression.ComboRegressionFormat.AddItem "x12save"
  regression.ComboRegressionAictest.Clear
  regression.ComboRegressionAictest.AddItem "trading day"
  regression.ComboRegressionAictest.AddItem "six td contrast variables"
  regression.ComboRegressionAictest.AddItem "stock trading day"

End Sub

Sub combo_series ()

  series.ComboSeriesPeriod.Clear
  series.ComboSeriesPeriod.AddItem "4"
  series.ComboSeriesPeriod.AddItem "12"
  series.ComboSeriesFormat.Clear
  series.ComboSeriesFormat.AddItem "free format"
  series.ComboSeriesFormat.AddItem "fortran format"
  series.ComboSeriesFormat.AddItem "1r"
  series.ComboSeriesFormat.AddItem "2r"
  series.ComboSeriesFormat.AddItem "1l"
  series.ComboSeriesFormat.AddItem "2l"
  series.ComboSeriesFormat.AddItem "cs"
  series.ComboSeriesFormat.AddItem "edit-date"
  series.ComboSeriesFormat.AddItem "x12save"
  series.ComboSeriesSpectrumtype.Clear
  series.ComboSeriesSpectrumtype.AddItem "periodogram"
  series.ComboSeriesSpectrumtype.AddItem "AR-spectrum"
  series.ComboSeriesComptype.Clear
```

```
    series.ComboSeriesComptype.AddItem "added"
    series.ComboSeriesComptype.AddItem "multiplicated"
    series.ComboSeriesComptype.AddItem "substracted"
    series.ComboSeriesComptype.AddItem "divided"

End Sub

Sub combo_sliding ()

    sliding.ComboSlidingOutlier.Clear
    sliding.ComboSlidingOutlier.AddItem "not performe outlier identification"
    sliding.ComboSlidingOutlier.AddItem "outlier identification in each estimation"
    sliding.ComboSlidingOutlier.AddItem "only carrie over any outlier"
    sliding.ComboSlidingFixmissing.Clear
    sliding.ComboSlidingFixmissing.AddItem "replace for each new estimation"
    sliding.ComboSlidingFixmissing.AddItem "use the original replacement value"
    sliding.ComboSlidingFixmdl.Clear
    sliding.ComboSlidingFixmdl.AddItem "parameters fixed and not reestimated"
    sliding.ComboSlidingFixmdl.AddItem "restore initial values and reestimate"
    sliding.ComboSlidingFixmdl.AddItem "default initial values and reestimate"

End Sub

Sub combo_transform ()

    transform.ComboTransformFunction.Clear
    transform.ComboTransformFunction.AddItem "none" 'poner el rango de valores de Yt
    transform.ComboTransformFunction.AddItem "log"
    transform.ComboTransformFunction.AddItem "sqrt"
    transform.ComboTransformFunction.AddItem "inverse"
    transform.ComboTransformFunction.AddItem "logistic"
    transform.ComboTransformAdjust.Clear
    transform.ComboTransformAdjust.AddItem "adjustment on monthly data"
    transform.ComboTransformAdjust.AddItem "adjustment on quarterly data"
    transform.ComboTransformAdjust.AddItem "no adjustment"
    transform.ComboTransformFormat.Clear
    transform.ComboTransformFormat.AddItem "free format"
    transform.ComboTransformFormat.AddItem "fortran format"
    transform.ComboTransformFormat.AddItem "1r"
    transform.ComboTransformFormat.AddItem "2r"
    transform.ComboTransformFormat.AddItem "1l"
    transform.ComboTransformFormat.AddItem "2l"
    transform.ComboTransformFormat.AddItem "cs"
    transform.ComboTransformFormat.AddItem "edit-date"
    transform.ComboTransformFormat.AddItem "x12save"
    transform.ComboTransformMode.Clear
    transform.ComboTransformMode.AddItem "factors in percents divided into the series"
    transform.ComboTransformMode.AddItem "factors in ratios divided into the series"
    transform.ComboTransformMode.AddItem "factors subtracted from the series"
    transform.ComboTransformApply.Clear
    transform.ComboTransformApply.AddItem "only in the modelling phase"
    transform.ComboTransformApply.AddItem "only in the seasonal adjustment phase"
    transform.ComboTransformApply.AddItem "in both phases"

End Sub

Sub combo_x11 ()

    x11.ComboX11Mode.Clear
    x11.ComboX11Mode.AddItem "multiplicative"
    x11.ComboX11Mode.AddItem "additive"
    x11.ComboX11Mode.AddItem "pseudo-additive"
    x11.ComboX11Mode.AddItem "log-additive"
    x11.ComboX11Seasonalma.Clear
    x11.ComboX11Seasonalma.AddItem "a 3x3 moving average"
    x11.ComboX11Seasonalma.AddItem "a 3x5 moving average"
    x11.ComboX11Seasonalma.AddItem "a 3x9 moving average"
    x11.ComboX11Seasonalma.AddItem "a 3x15 moving average"
    x11.ComboX11Seasonalma.AddItem "stable seasonal filter"
    x11.ComboX11Seasonalma.AddItem "initialy 3x3 and finally 3x5"
    x11.ComboX11Td.Clear
    x11.ComboX11Td.AddItem "apply"
    x11.ComboX11Td.AddItem "significant"
```

```
  x11.ComboX11Td.AddItem "noapply"
  x11.ComboX11Td.AddItem "prior"
  x11.ComboX11Td.AddItem "priorsignificant"
  x11.ComboX11Final.Clear
  x11.ComboX11Final.AddItem "prior adjustment factors"
  x11.ComboX11Final.AddItem "holiday adjustment factors"
  x11.ComboX11Final.AddItem "no remove adjustment factors"
  x11.ComboX11Lom.Clear
  x11.ComboX11Lom.AddItem "include in the trading day component"
  x11.ComboX11Lom.AddItem "include in the seasonal component"
  x11.ComboX11Calendarsigma.Clear
  x11.ComboX11Calendarsigma.AddItem "default"
  x11.ComboX11Calendarsigma.AddItem "computed separately for each month (quarter)"
  x11.ComboX11Calendarsigma.AddItem "computed separately only if Cochran's"
  x11.ComboX11Calendarsigma.AddItem "computed separately for each grup"
  x11.ComboX11Shortsf.Clear
  x11.ComboX11Shortsf.AddItem "stable seasonal filter"
  x11.ComboX11Shortsf.AddItem "seasonal filter selected before"
  x11.ComboX11Extremeadj.Clear
  x11.ComboX11Extremeadj.AddItem "sample standar deviation"
  x11.ComboX11Extremeadj.AddItem "weighted median absolute deviation"
  x11.ComboX11Extremeadj.AddItem "exponential of weighted median absolute deviation of the logs"
  x11.ComboX11Force.Clear
  'explicarlo más en el help
  x11.ComboX11Force.AddItem "same yearly totals"
  x11.ComboX11Force.AddItem "same rounded totals"
  x11.ComboX11Force.AddItem "first yearly totals and then rounded total"

End Sub

Function Ctovbstring$ (InString$)

  Ctovbstring = Mid$(InString$, 1, Len(InString$) - I)

End Function

Sub descargar ()

  Unload arima
  Unload automdl
  Unload check
  Unload composite
  Unload estimate
  Unload forecast
  Unload history
  Unload identify
  Unload outlier
  Unload regadjust
  Unload regression
  Unload series
  Unload sliding
  Unload transform
  Unload x11
  'Unload errorform
  'Unload outputform
  'Unload graphvalues
  'Unload plot
  'Unload tables
  'Unload formx12
End Sub

Sub inicializar ()
Dim X%

  title$ = "X-12-ARIMA"

  Call titleform

  'periodicidad$ = series.ComboSeriesPeriod.List(series.ComboSeriesPeriod.ListIndex)
  'start$ = series.TextSeriesStart.Text
  If (getvar("series", "period")) = 0 Then
    periodicidad$ = "4"
  End If
```

```
  If (getvar("series", "period")) = 1 Then
    periodicidad$ = "12"
  End If
  start$ = (getvar("series", "start"))

  Unload series

  'transform.TextTransformStart.Text = start$
  'outlier.TextOutlierSpan(0).Text = start$
  'series.TextSeriesSpan(0).Text = start$
  'series.TextSeriesModelspan(0).Text = start$
  'composite.TextCompositeModelspan(0).Text = start$
  'x11.TextX11Tdstart.Text = Left$(start$, 4)
  'x11.TextX11Tdapply.Text = Left$(start$, 4)
  col% = 0

  'me parece que de aquí para abajo no sirve para nada
  For X% = 1 To 15
    checkcargar(X%) = "no"
    Call setvar("formx12", "arguments" & X% - 1, False)
  Next X%

  Call setvar("series", "cargar", checkcargar(1))
  Call setvar("composite", "cargar", checkcargar(2))
  Call setvar("transform", "cargar", checkcargar(3))
  Call setvar("x11", "cargar", checkcargar(4))
  Call setvar("identify", "cargar", checkcargar(5))
  Call setvar("regression", "cargar", checkcargar(6))
  Call setvar("arima", "cargar", checkcargar(7))
  Call setvar("automdl", "cargar", checkcargar(8))
  Call setvar("estimate", "cargar", checkcargar(9))
  Call setvar("outlier", "cargar", checkcargar(10))
  Call setvar("check", "cargar", checkcargar(11))
  Call setvar("forecast", "cargar", checkcargar(12))
  Call setvar("regadjust", "cargar", checkcargar(13))
  Call setvar("sliding", "cargar", checkcargar(14))
  Call setvar("history", "cargar", checkcargar(15))
  'hasta aquí no sirve para nada


End Sub

Sub inicializar_arima ()
Dim X%

  'mirar Call setvar("arima", "fix", 0)
  Call setvar("arima", "title", "ARIMA model")
  For X% = 0 To 5   'inicializar hasta x%=14
    Call setvar("arima", "model" & X%, 0)
  Next X%

End Sub

Sub inicializar_automdl ()
Dim X%, numfichero%, texto$, Ret$

  Ret$ = Chr(13) & Chr(10)
  'cargar los listindex de combo
  Call setvar("automdl", "comboautomdl0", 0)
  Call setvar("automdl", "comboautomdl1", 1)
  Call setvar("automdl", "comboautomdl2", 1)
  Call setvar("automdl", "comboautomdl3", 1)

  'no hace falta el file
  Call setvar("automdl", "textlimit0", 15)
  Call setvar("automdl", "textlimit1", 20)
  Call setvar("automdl", "textlimit2", 5)
  Call setvar("automdl", "textlimit3", "0.9")
  Call setvar("automdl", "maxmod", 5)
  Call setvar("automdl", "models0", "(0 1 1) (0 1 1)")
  Call setvar("automdl", "models1", "(0 1 2) (0 1 1)")
  Call setvar("automdl", "models2", "(2 1 0) (0 1 1)")
  Call setvar("automdl", "models3", "(0 2 2) (0 1 1)")
```

```
  Call setvar("automdl", "models4", "(2 1 2) (0 1 1)")
  texto = ""
  For X% = 0 To 3
    texto = texto & getvar("automdl", "models" & X%) & " X" & Ret
  Next X%
  texto = texto & getvar("automdl", "models4") & Ret
  numfichero = FreeFile
  Open app.Path & "\output\modelo.mdl" For Output As 1
  Print #1, texto
  Close 1


End Sub

Sub inicializar_composite ()
Dim X%

  Call setvar("composite", "textdecimals0", 0)
  'composite.textcompositespectrumstart.text= ??
  Call setvar("composite", "textdecimals1", 15)
  'cargar listindex de combo
  Call setvar("composite", "spectrumtype", 1)
  Call setvar("composite", "textcomposite0", start$)
  Call setvar("composite", "diffspectrum0", True)
  Call setvar("composite", "diffspectrum1", False)
  For X% = 0 To 7
    Call setvar("composite", "save" & X%, True)
  Next X%
  For X% = 8 To 11
    Call setvar("composite", "save" & X%, False)
  Next X%
  For X% = 12 To 16
    Call setvar("composite", "save" & X%, True)
  Next X%
  For X% = 17 To 18
    Call setvar("composite", "save" & X%, False)
  Next X%

End Sub

Sub inicializar_check ()
Dim X%

  Call setvar("check", "textmaxlag", 36)
  For X% = 0 To 1
    Call setvar("check", "save" & X%, True)
  Next X%

End Sub

Sub inicializar_estimate ()
Dim X%

  'cargar listindex de combo
  Call setvar("estimate", "parms", 0)
  Call setvar("estimate", "exact", 0)
  Call setvar("estimate", "outofsample", 1)

  Call setvar("estimate", "tol", "0.00001")
  Call setvar("estimate", "textmaxiter", 200)
  Call setvar("estimate", "checkestimate0", True)
  Call setvar("estimate", "checkestimate1", False)
  Call setvar("estimate", "checkestimate2", True)

End Sub

Sub inicializar_forecast ()
Dim X%

  Call setvar("forecast", "textforecast0", periodicidad$)
  Call setvar("forecast", "textforecast1", 0)
  Call setvar("forecast", "textforecast2", 0)
  Call setvar("forecast", "textforecast3", "0.95")
```

```
 For X% = 0 To 2
  Call setvar("forecast", "save" & X%, True)
 Next X%

End Sub

Sub inicializar_history ()
Dim X%

 'cargar listindex de combo
 Call setvar("history", "combotype", 1)
 Call setvar("history", "refresh", 1)
 Call setvar("history", "adjfcst", 0)
 Call setvar("history", "outlier", 0)

 Call setvar("history", "revisionlag", 1) 'mirar pq el cero no funciona
 Call setvar("history", "fstep0", 1)
 Call setvar("history", "fstep1", periodicidad$)
 Call setvar("history", "outlierwin", periodicidad$)
 Call setvar("history", "checkestimates0", True)
 Call setvar("history", "checkhistory0", False)
 Call setvar("history", "checkhistory1", True)
 Call setvar("history", "checkhistory2", False)
 For X% = 1 To 7
  Call setvar("history", "checkestimates" & X%, False)
 Next X%

End Sub

Sub inicializar_identify ()

 Call setvar("identify", "text0", 0)
 Call setvar("identify", "text1", 0)
 Call setvar("identify", "text2", 36)

End Sub

Sub inicializar_outlier ()
Dim X%

 'cargar listindex de combo
 Call setvar("outlier", "types", 2)
 Call setvar("outlier", "method", 0)

 For X% = 0 To 1
  Call setvar("outlier", "critical" & X%, "3.3")
 Next X%
 Call setvar("outlier", "textlsrun", 0)
 Call setvar("outlier", "span0", start$)

End Sub

Sub inicializar_regadjust ()
Dim X%

 'cargar listindex de combo
 Call setvar("regadjust", "augmentusertd", 0)
 Call setvar("regadjust", "listchoices0", "constant") 'n
 Call setvar("regadjust", "listchoices1", "td") 'n
 Call setvar("regadjust", "listchoices2", "ao")
 Call setvar("regadjust", "listchoices3", "ls")
 Call setvar("regadjust", "listchoices4", "rp") 'n
 Call setvar("regadjust", "listchoices5", "holiday") 'n
 Call setvar("regadjust", "listchoices6", "seasonal")
 Call setvar("regadjust", "listchoices7", "stocklom")
 Call setvar("regadjust", "listchoices8", "lom") 'n
 Call setvar("regadjust", "listchoices9", "loq") 'n
 Call setvar("regadjust", "listchoices10", "lpyear") 'n
 Call setvar("regadjust", "listchoices11", "user") 'n
 Call setvar("regadjust", "optionuser0", True)


End Sub
```

```
Sub inicializar_regression ()
Dim X%

  'cargar listindex de combo
  Call setvar("regression", "comboformat", 0)
  Call setvar("regression", "comboaictest", 0)

  Call setvar("regression", "start", start$)

End Sub

Sub inicializar_series ()
Dim X%

  Call setvar("series", "start", "0001.01")
  For X% = 0 To 1
    Call setvar("series", "textseries" & X%, 0)
  Next X%
  Call setvar("series", "compwt", 1)
  Call setvar("series", "missingcode", -9999)
  Call setvar("series", "missingval", 100000000)
  Call setvar("series", "textseries2", 15)
  ' mirar Call setvar("series", "spectrumstart", "0001.01")
  'cargar listindex de combos
  Call setvar("series", "period", 1)
  Call setvar("series", "comboformat", 0)
  Call setvar("series", "spectrumtype", 1)
  Call setvar("series", "diffspectrum", True)
  Call setvar("series", "comptype", 0)
  For X% = 0 To 1
    Call setvar("series", "save" & X%, True)
  Next X%

End Sub

Sub inicializar_sliding ()
Dim X%

  'cargar listindex de combo
  Call setvar("sliding", "outlier", 0)
  Call setvar("sliding", "fixmissing", 1)
  Call setvar("sliding", "fixmdl", 0)

  Call setvar("sliding", "cutseas", "3.0")
  Call setvar("sliding", "cutchng", "3.0")
  Call setvar("sliding", "cuttd", "2.0")

End Sub

Sub inicializar_transform ()
Dim X%

    Call setvar("transform", "start", "0001.01")
    Call setvar("transform", "textprecision", 0)
    Call setvar("transform", "power", 1)
    'cargar listindex de combo
    Call setvar("transform", "function", 0)
    Call setvar("transform", "adjust", 2)
    Call setvar("transform", "comboformat", 0)
    Call setvar("transform", "mode", 0)
    Call setvar("transform", "apply", 2)
    For X% = 0 To 1
      Call setvar("transform", "save" & X%, False)
    Next X%
    Call setvar("transform", "save2", True)
    Call setvar("transform", "save3", False)

End Sub

Sub inicializar_x11 ()
Dim X%
```

```
  'cargar listindex de combo
  Call setvar("x11", "mode", 0)
  Call setvar("x11", "seasonalma", 0)
  Call setvar("x11", "combox11", 0)
  Call setvar("x11", "final", 2)
  Call setvar("x11", "combolom", 1)
  Call setvar("x11", "calendarsigma", 0)
  Call setvar("x11", "shortsf", 0)
  Call setvar("x11", "extremeadj", 0)
  Call setvar("x11", "force", 0)

  Call setvar("x11", "texttrendma", 3)
  Call setvar("x11", "sigmalim0", "1.5")
  Call setvar("x11", "sigmalim1", "2.5")
  'Call setvar("x11", "tdstart", "0001.01")
  'call setvar("x11","tdapply","0001.01")
  Call setvar("x11", "tdexclude", "2.5")
  For X% = 0 To 6
    Call setvar("x11", "tdprior" & X%, "1.0")
  Next X%
  Call setvar("x11", "texttdmaxlead", periodicidad$)
  Call setvar("x11", "taper", 1)
  Call setvar("x11", "itrendma", 1)
  Call setvar("x11", "seasonalma0", True)
  Call setvar("x11", "seasonalma1", False)
  Call setvar("x11", "seasonalma2", False)
  Call setvar("x11", "type", True) 'n
  For X% = 0 To 61
    Call setvar("x11", "checkx11" & X%, False)
  Next X%
  Call setvar("x11", "checkx112", True)
  For X% = 8 To 10
    Call setvar("x11", "checkx11" & X%, True)
  Next X%
  For X% = 13 To 14
    Call setvar("x11", "checkx11" & X%, True)
  Next X%
  Call setvar("x11", "checkx1116", True)
  Call setvar("x11", "checkx1135", True)
  For X% = 55 To 56
    Call setvar("x11", "checkx11" & X%, True)
  Next X%

End Sub

Sub periodo ()

    Select Case periodicidad$
      Case "12"
        regression.Check3DRegressionVariables(6).Enabled = False
        regression.Check3DRegressionVariables(6).Value = False
        regression.Check3DRegressionVariables(6).FontBold = True
        If transform.ComboTransformAdjust.ListIndex = 1 Then
          response% = MsgBox("The adjustment must be on monthly data", 48, title$)
          transform.ComboTransformAdjust.SetFocus
        End If
      Case "4"
        regression.Check3DRegressionVariables(5).Enabled = False
        regression.Check3DRegressionVariables(5).Value = False
        regression.Check3DRegressionVariables(5).FontBold = True
        If transform.ComboTransformAdjust.ListIndex = 0 Then
          response% = MsgBox("The adjustment must be on quarterly data", 48, title$)
          transform.ComboTransformAdjust.SetFocus
        End If
    End Select

End Sub

Function Strtran$ (ByVal in$, char1$, char2$)
Dim X&
Dim out$
Dim char$
Dim charlen%
```

```
Dim centinela%

 charlen% = Len(char1$)

 For X& = 1 To Len(in$)
  char$ = Mid$(in$, X&, charlen%)
  If char$ = char1$ Then
   out$ = out$ & char2$
   centinela% = charlen%
  ElseIf centinela <= 0 Then
   out$ = out$ & Left$(char$, 1)
  End If
  centinela% = centinela% - 1
 Next X&

 Strtran$ = out$

End Function

Sub titleform ()

   titlemenu$ = "ISAX (" & LCase(X12IniFile$) & ")"
   titleformx12$ = "ISAX Configuration (" & LCase(X12IniFile$) & ")"
   titlegraph$ = "Graphics (" & LCase(X12IniFile$) & ")"
   titleerror$ = "Error (" & LCase(X12IniFile$) & ")"
   titleoutput$ = "Output (" & LCase(X12IniFile$) & ")"
   titlegraphval$ = "Values (" & LCase(X12IniFile$) & ")"
   titleplot$ = "Plot (" & LCase(X12IniFile$) & ")"
   titletables$ = "Graphics (" & LCase(X12IniFile$) & ")"

End Sub

Function validar (tecla As Integer)
Dim c As String * 1

   c = Chr(tecla)
   If (c < "0" Or c > "9") And c <> Chr(13) And c <> Chr(8) And c <> "." And c <> "-" And c <> " " Then
      validar = 0
   Else
      validar = tecla
   End If

End Function

Sub validar_arguments ()
Dim i%

'Select Case i
 'Case 0 'series
 'If formx12.Arguments(0).Checked = True Then
 '  If formx12.Arguments(1).Checked = True Then
 '    response% = MsgBox("Series cannot be used with composite", 48, title$)
 '    formx12.Arguments(0).Checked = False
 '  End If
 'End If


 'Case 1 'composite
 'Case 2 'transform
 'Case 3 'x11
 'Case 4 'identify
 'Case 5 'regression
 'Case 6 'arima
 'Case 7 'automdl
 'Case 8 'estimate
 'Case 9 'outlier
 'Case 10 'check
 'Case 11 'forecast
 'Case 12 'regadjust
 'Case 13 'slidingspans
 'Case 14 'history
'End Select
```

```
End Sub
```

1.2.2. X12Comun.bas

```
Option Explicit

Function Cargar_datos1% (fn%, col%)
Dim ch$, r%, aux$

   ch$ = Input$(1, fn%)
   r% = 0
   aux$ = ""
   'eliminar las dos primeras filas
   While ch$ <> Chr(13)
      ch$ = Input$(1, fn%)
   Wend
   ch$ = Input$(1, fn%)
   While ch$ <> Chr(13)
      ch$ = Input$(1, fn%)
   Wend
   ch$ = Input$(1, fn%)
   'leer los datos

   While Not EOF(fn%)
      While ch$ <> Chr(9) And Not EOF(fn%)
         ch$ = Input$(1, fn%)
      Wend
      While ch$ <> Chr(13) And Not EOF(fn%)
         ch$ = Input$(1, fn%)
         If ch$ <> Chr(13) Then
            aux$ = aux$ & ch$
         End If
      Wend
      'ReDim Preserve GraphArray(25, r)
      'ReDim Preserve grapharray(col, r)
      GraphArray(col%, r) = Val(aux$)
      r = r + 1
      aux$ = ""
   Wend

 Cargar_datos1% = r%

End Function

Sub execute_0 (Serie$)
Dim x%, ins$, specname$, f$


   'crear directorio output
   If Dir$(app.Path & "\output", 18) <> "" Then
      'If Dir$(app.Path & "\output\" & series_name(CStr(series!labelseriesfile.caption)) & ".*") <> "" Then
      If Dir$(app.Path & "\output\" & series_name(CStr((getvar("series", "file")))) & ".*") <> "" Then
         'Kill app.Path & "\output\" & series_name(CStr(series!labelseriesfile.caption)) & ".*"
         Kill app.Path & "\output\" & series_name(CStr((getvar("series", "file")))) & ".*"
      End If
      Else
         MkDir app.Path & "\output"
   End If

   Call spec_file

   'f$ = Mid$(menuprincipal.CMDialogSpec.Filename, 1, Len(menuprincipal.CMDialogSpec.Filename) - 4)
   'f$ = app.Path & "\output\" & series_name(CStr(seriesname))
   f$ = Mid$(Serie$, 1, Len(Serie$) - 4)

   If Dir$(f$ & ".x2d") <> "" Then Kill f$ & ".x2d"
   If Dir$(f$ & ".xdg") <> "" Then Kill f$ & ".xdg"

   ins$ = app.Path & "\x12a.pif " & f$ & " -n -p -s -o " & app.Path & "\output\" & series_name(CStr(f$))
   'menuprincipal.Command3D(2).Enabled = False
   x% = Shell(ins$, 2)
   Do Until getmoduleusage(x%) = 0
     DoEvents
   Loop
   'menuprincipal.Command3D(2).Enabled = True
```

```
    'If Dir$(f$ & ".xdg") <> "" Then
    '   x% = Shell(app.Path & "\x12diag.pif " & f$ & " -o " & app.Path & "\output\" & series_name(CStr(f$)), 2)
    '   Do Until getmoduleusage(x%) = 0 'n
    '      DoEvents 'n
    '   Loop 'n
    'End If

End Sub

Function getvar (Carpeta$, VarName$)
Dim buff$, x%

  buff$ = Space(50)
  x% = OsGetPrivateProfileString(Carpeta$, VarName$, "", buff$, Len(buff$), X12IniFile$)
  'getvar = Val(Strtran$(Ctovbstring(Trim(buff$)), ",", "."))
  getvar = (Strtran$(Ctovbstring(Trim(buff$)), ",", "."))

End Function

Function series_name$ (Seriesfile$)
Dim var1$, c As String * 1, i%

    If Mid$(Seriesfile, Len(Seriesfile) - 3, 1) = "." Then
       var1 = Left$(Seriesfile, Len(Seriesfile) - 4)
       i = Len(var1)
       While c <> "\" And i > 1
       i = i - 1
       c = Mid$(var1, i, 1)
       Wend
       If c = "\" Then
       series_name = Mid(var1, i + 1, Len(var1) - i)
       Else
          series_name = var1
       End If

       Else

        If Mid$(Seriesfile, Len(Seriesfile) - 2, 1) = "." Then
          var1 = Left$(Seriesfile, Len(Seriesfile) - 3)
          i = Len(var1)
          While c <> "\" And i > 1
            i = i - 1
            c = Mid$(var1, i, 1)
          Wend
          If c = "\" Then
            series_name = Mid(var1, i + 1, Len(var1) - i)
            Else
              series_name = var1
          End If

          Else

            If Mid$(Seriesfile, Len(Seriesfile) - 1, 1) = "." Then
              var1 = Left$(Seriesfile, Len(Seriesfile) - 2)
              i = Len(var1)
              While c <> "\" And i > 1
                i = i - 1
                c = Mid$(var1, i, 1)
              Wend
              If c = "\" Then
                series_name = Mid(var1, i + 1, Len(var1) - i)
                Else
                  series_name = var1
              End If

              Else
                var1 = Seriesfile
                i = Len(var1)
                While c <> "\" And i > 1
                  i = i - 1
                  c = Mid$(var1, i, 1)
                Wend
                If c = "\" Then
```

```
            series_name = Mid(var1, i + 1, Len(var1) - i)
            Else
              series_name = var1
           End If
         End If
       End If
  End If

End Function

Sub setvar (Carpeta$, VarName$, VarValue As Variant)
Dim x%

 x% = OSWritePrivateProfileString(Carpeta$, VarName$, VarValue, X12IniFile$)

End Sub

Sub spec_file ()
Dim texto$
Dim numfichero%

  'descargar
  Tabul$ = Chr(9)
  CRLF$ = Chr(13) & Chr(10)
  texto = ""

  If (getvar("formx12", "arguments0")) = True Then
    texto = texto & series_text()
  End If
  If (getvar("formx12", "arguments1")) = True Then
    texto = texto & composite_text()
  End If
  If (getvar("formx12", "arguments2")) = True Then
    texto = texto & transform_text()
  End If
  If (getvar("formx12", "arguments3")) = True Then
    texto = texto & x11_text()
  End If
  If (getvar("formx12", "arguments4")) = True Then
    texto = texto & identify_text()
  End If
  If (getvar("formx12", "arguments5")) = True Then
    texto = texto & regression_Text()
  End If
  If (getvar("formx12", "arguments6")) = True Then
    texto = texto & arima_text()
  End If
  If (getvar("formx12", "arguments7")) = True Then
    texto = texto & automdl_Text()
  End If
  If (getvar("formx12", "arguments8")) = True Then
    texto = texto & estimate_text()
  End If
  If (getvar("formx12", "arguments9")) = True Then
    texto = texto & outlier_Text()
  End If
  If (getvar("formx12", "arguments10")) = True Then
    texto = texto & check_Text()
  End If
  If (getvar("formx12", "arguments11")) = True Then
    texto = texto & forecast_text()
  End If
  If (getvar("formx12", "arguments12")) = True Then
    texto = texto & regadjust_Text()
  End If
  If (getvar("formx12", "arguments13")) = True Then
    texto = texto & slidingspans_text()
  End If
  If (getvar("formx12", "arguments14")) = True Then
    texto = texto & history_text()
  End If

  numfichero = FreeFile
```

```
  'menuprincipal.CMDialogSpec.Action = 1
  'Open menuprincipal.CMDialogSpec.Filename For Output As #1
  Select Case runmode%
   Case 0 'single mode (one series)
    seriesname = app.Path & "\output\" & series_name(CStr(getvar("series", "file"))) & ".spc"
   Case 1 'multiple mode (some series one configuration)
    seriesname = app.Path & "\output\metafile.spc"
   Case 2 'multiple mode (some series some configurations)
    seriesname = app.Path & "\output\" & series_name(CStr(getvar("series", "file"))) & ".spc"
   Case 3 'composite
    If Val(getvar("formx12", "arguments0")) <> 0 Then
     seriesname = app.Path & "\output\" & series_name(CStr(getvar("series", "file"))) & ".spc"
    End If
    If Val(getvar("formx12", "arguments1")) <> 0 Then
     seriesname = app.Path & "\output\" & (getvar("composite", "name")) & ".spc"
    End If
  End Select

  'If (getvar("series", "file")) <> "" Then
  '   seriesname = app.Path & "\output\" & series_name(CStr(getvar("series", "file"))) & ".spc"
  ' Else seriesname = app.Path & "\output\metafile.spc"
  'End If

  Open seriesname For Output As 1
  Print #1, texto
  Close #1

End Sub
```

1.2.3. X12Specf.bas

```
Option Explicit
Global seriesname$


Function arima_text$ ()
Dim texto$, x%
Dim seaCnt%, paramCnt%

texto = ""
'If arima.Check3DArima.Value = False Then para recuperar un modelo
'aún no se puede hacer. No funciona x12
  texto = texto & "arima{" & CRLF$ & Tabul$
  'If arima.Check3DArima.Value = False Then
    texto = texto & "model=("
    If Len(getvar("arima", "model0")) > 1 Then
      texto = texto & "[" & (getvar("arima", "model0")) & "] "
      Else
        texto = texto & (getvar("arima", "model0")) & " "
    End If
    texto = texto & (getvar("arima", "model1")) & " "
    If Len(getvar("arima", "model2")) > 1 Then
      texto = texto & "[" & (getvar("arima", "model2")) & "] "
      Else
        texto = texto & (getvar("arima", "model2")) & " "
    End If
    texto = texto & ")"

    'For x% = 0 To 2
    '  texto = texto & Val(getvar("arima", "model" & x%)) & " "
    'Next x%

    'texto = texto & ")" & "("
    seaCnt% = 1
    paramCnt% = 3
    Do While Val(getvar("arima", "seaso" & seaCnt%)) <> 0
      texto = texto & "("

      texto = texto & Val(getvar("arima", "model" & paramCnt%)) & " "
      paramCnt% = paramCnt% + 1
      texto = texto & Val(getvar("arima", "model" & paramCnt%)) & " "
      paramCnt% = paramCnt% + 1
      texto = texto & Val(getvar("arima", "model" & paramCnt%))
      paramCnt% = paramCnt% + 1

      texto = texto & ")" & getvar("arima", "seaso" & seaCnt%)
      seaCnt% = seaCnt% + 1
    Loop

    texto = texto & CRLF$ & Tabul$

    If (getvar("arima", "checklistar")) = "yes" Then 'lista no vacía
      texto = texto & "ar=("
      For x% = 0 To Val(getvar("arima", "maxar")) - 1
        texto = texto & (getvar("arima", "arparam" & x%)) & ","
      Next x%
      texto = texto & (getvar("arima", "arparam" & Val(getvar("arima", "maxar")))) & ")"
      texto = texto & CRLF$ & Tabul$
    End If
    If (getvar("arima", "checklistma")) = "yes" Then 'lista no vacía
      texto = texto & "ma=("
      For x% = 0 To Val(getvar("arima", "maxma")) - 1
        texto = texto & (getvar("arima", "maparam" & x%)) & ","
      Next x%
      texto = texto & (getvar("arima", "maparam" & Val(getvar("arima", "maxma")))) & ")"
      texto = texto & CRLF$ & Tabul$
    End If
  'End If
  'If arima.Check3DArima.Value = True Then
  '  texto = texto & "file='" & arima.TextArimaFile.Text & "'"
  '  texto = texto & CRLF$ & Tabul$
```

```
'End If
texto = texto & "title='" & (getvar("arima", "title")) & "'"
texto = texto & CRLF$ 'no tab
texto = texto & "}" & CRLF$ & CRLF$ 'no tab
'Else
' Open "c:\x12\serie.mdl" For Input As 1
' texto = texto & Input(LOF(1), 1)
' Close 1
' 'texto=texto & CRLF$ & Tabul$
' 'texto = texto & "fix="
' 'Select Case arima.ComboArimaFix.ListIndex
' ' Case 0
' '   texto = texto & "both"
' ' Case 1
' '   texto = texto & "arma"
' ' Case 2
' '   texto = texto & "none"
' 'End Select
' texto = texto & CRLF$ & CRLF$
'End If

arima_text = texto

End Function

Function automdl_Text$ ()
Dim texto$

texto = ""
texto = texto & "automdl{" & CRLF$ & Tabul$
'mode
texto = texto & "method="
Select Case Val(getvar("automdl", "comboautomdl1"))
  Case 0
    texto = texto & "first"
  Case 1
    texto = texto & "best"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "file='"
texto = texto & app.Path & "\output\modelo.mdl'"
'texto = texto & "file='" & automdl!TextAutomdlFile.Text & "'"
texto = texto & CRLF$ & Tabul$
texto = texto & "fcstlim=" & (getvar("automdl", "textlimit0"))
texto = texto & CRLF$ & Tabul$
texto = texto & "bcstlim=" & (getvar("automdl", "textlimit1"))
texto = texto & CRLF$ & Tabul$
texto = texto & "qlim=" & (getvar("automdl", "textlimit2"))
texto = texto & CRLF$ & Tabul$
texto = texto & "overdiff=" & (getvar("automdl", "textlimit3"))
texto = texto & CRLF$ & Tabul$
texto = texto & "identify="
Select Case Val(getvar("automdl", "comboautomdl2"))
  Case 0
    texto = texto & "all"
  Case 1
    texto = texto & "first"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "outofsample="
Select Case Val(getvar("automdl", "comboautomdl3"))
  Case 0
    texto = texto & "yes"
  Case 1
    texto = texto & "no"
End Select
texto = texto & CRLF$ & CRLF$ & Tabul$
texto = texto & "print=(all)"
texto = texto & CRLF$ 'no tab
texto = texto & "}" & CRLF$ & CRLF$'no tab

automdl_Text = texto
```

```
End Function

Function composite_text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "composite{"
  texto = texto & CRLF$ & Tabul$
  texto = texto & "name=" & (getvar("composite", "name"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "decimals=" & (getvar("composite", "textdecimals0"))
  texto = texto & CRLF$ & Tabul$
  If Val(getvar("composite", "checkmodelspan")) <> 0 Then
     texto = texto & "modelspan=("
     texto = texto & (getvar("composite", "textcomposite0")) & "," & (getvar("composite", "textcomposite1"))
     texto = texto & ")" & CRLF$ & Tabul$
  End If
  If (getvar("composite", "spectrumstart")) <> "" Then
   texto = texto & "startspectrum=" & (getvar("composite", "spectrumstart"))
   texto = texto & CRLF$ & Tabul$
  End If
  texto = texto & "spectrumtype="
  Select Case Val(getvar("composite", "spectrumtype"))
     Case 0
        texto = texto & "periodogram"
     Case 1
        texto = texto & "arspec"
  End Select
  texto = texto & CRLF$ & Tabul$
  texto = texto & "diffspectrum="
  If Val(getvar("composite", "diffspectrum0")) <> 0 Then
     texto = texto & "yes"
  End If
  If Val(getvar("composite", "diffspectrum1")) <> 0 Then
     texto = texto & "no"
  End If
  texto = texto & CRLF$ & Tabul$
  texto = texto & "saveprecision=" & (getvar("composite", "textdecimals1"))
  texto = texto & CRLF$ & Tabul$
  For i = 0 To 18
     If Val(getvar("composite", "save" & i%)) <> 0 Then
        a = "true"
        Exit For
     End If
  Next i%
  If a = "true" Then
     texto = texto & "save=("
     If Val(getvar("composite", "save0")) <> 0 Then
        texto = texto & "cms "
     End If
     If Val(getvar("composite", "save1")) <> 0 Then
        texto = texto & "id8 "
     End If
     If Val(getvar("composite", "save2")) <> 0 Then
        texto = texto & "isf "
     End If
     If Val(getvar("composite", "save3")) <> 0 Then
        texto = texto & "isa "
     End If
     If Val(getvar("composite", "save4")) <> 0 Then
        texto = texto & "iaa "
     End If
     If Val(getvar("composite", "save5")) <> 0 Then
        texto = texto & "irn "
     End If
     If Val(getvar("composite", "save6")) <> 0 Then
        texto = texto & "itn "
     End If
     If Val(getvar("composite", "save7")) <> 0 Then
        texto = texto & "iir "
     End If
     If Val(getvar("composite", "save8")) <> 0 Then
        texto = texto & "ie1 "
```

```
      End If
      If Val(getvar("composite", "save9")) <> 0 Then
         texto = texto & "ie2 "
      End If
      If Val(getvar("composite", "save10")) <> 0 Then
         texto = texto & "ie3 "
      End If
      If Val(getvar("composite", "save11")) <> 0 Then
         texto = texto & "ie4 "
      End If
      If Val(getvar("composite", "save12")) <> 0 Then
         texto = texto & "ie5 "
      End If
      If Val(getvar("composite", "save13")) <> 0 Then
         texto = texto & "ie6 "
      End If
      If Val(getvar("composite", "save14")) <> 0 Then
         texto = texto & "i6a "
      End If
      If Val(getvar("composite", "save15")) <> 0 Then
         texto = texto & "i6r "
      End If
      If Val(getvar("composite", "save16")) <> 0 Then
         texto = texto & "ie7 "
      End If
      If Val(getvar("composite", "save17")) <> 0 Then
         texto = texto & "iee "
      End If
      If Val(getvar("composite", "save18")) <> 0 Then
         texto = texto & "if1 "
      End If
      texto = texto & ")" & CRLF$ 'no tab
   End If
   texto = texto & "}" & CRLF$ & CRLF$'no tab

   composite_text = texto


End Function

Function check_Text$ ()
Dim texto$, i%, a$

   texto = ""
   texto = texto & "check{" & CRLF$ & Tabul$
   texto = texto & "maxlag=" & (getvar("check", "textmaxlag"))
   texto = texto & CRLF$ & Tabul$
   For i% = 0 To 1
      If Val(getvar("check", "save" & i%)) <> 0 Then
         a = "true"
         Exit For
      End If
   Next i%
   If a = "true" Then
      texto = texto & "save=("
      If Val(getvar("check", "save0")) <> 0 Then
         texto = texto & "acf "
      End If
      If Val(getvar("check", "save1")) <> 0 Then
         texto = texto & "pacf "
      End If
      texto = texto & ")" & CRLF$'no tab
   End If
   'quitar la fila entera
   texto = texto & "print=(acfplot,pacfplot)" & CRLF$
   texto = texto & "}" & CRLF$ & CRLF$

   check_Text = texto

End Function

Function estimate_text$ ()
Dim texto$, i%, a$
```

```
texto = ""
texto = texto & "estimate{" & CRLF$ & Tabul$
texto = texto & "parms="
Select Case Val(getvar("estimate", "parms"))
  Case 0
    texto = texto & "estimated"
  Case 1
    texto = texto & "fixed"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "tol=" & (getvar("estimate", "tol"))
texto = texto & CRLF$ & Tabul$
texto = texto & "maxiter=" & (getvar("estimate", "textmaxiter"))
texto = texto & CRLF$ & Tabul$
texto = texto & "exact="
Select Case Val(getvar("estimate", "exact"))
  Case 0
    texto = texto & "arma"
  Case 1
    texto = texto & "ma"
  Case 2
    texto = texto & "none"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "outofsample="
Select Case Val(getvar("estimate", "outofsample"))
  Case 0
    texto = texto & "yes"
  Case 1
    texto = texto & "no"
End Select
texto = texto & CRLF$ & Tabul$
For i% = 0 To 2
  If Val(getvar("estimate", "checkestimate" & i%)) <> 0 Then
    a = "true"
    Exit For
  End If
Next i%
If a = "true" Then
  texto = texto & "save=("
  'If estimate!Check3DEstimate(0).Value = True Then
  '  texto = texto & "iterations "
  'End If
  'If estimate!Check3DEstimate(1).Value = True Then
  '  texto = texto & "model "
  'End If
  'If estimate!Check3DEstimate(2).Value = True Then
  '  texto = texto & "regcmatrix "
  'End If
  'If estimate!Check3DEstimate(3).Value = True Then
  '  texto = texto & "armacmatrix "
  'End If
  'If estimate!Check3DEstimate(4).Value = True Then
  '  texto = texto & "estimates "
  'End If
  'If estimate!Check3DEstimate(5).Value = True Then
  '  texto = texto & "lkstats "
  'End If
  'If estimate!Check3DEstimate(6).Value = True Then
  '  texto = texto & "lkstats "
  'End If
  'If estimate!Check3DEstimate(7).Value = True Then
  '  texto = texto & "roots "
  'End If
  'If estimate!Check3DEstimate(8).Value = True Then
  '  texto = texto & "regressioneffects "
  'End If
  'If estimate!Check3DEstimate(9).Value = True Then
  '  texto = texto & "residuals "
  'End If
  If Val(getvar("estimate", "checkestimate0")) <> 0 Then
    texto = texto & "residuals "
```

```
    End If
    If Val(getvar("estimate", "checkestimate1")) <> 0 Then
      texto = texto & "model "
    End If
    If Val(getvar("estimate", "checkestimate2")) <> 0 Then
      texto = texto & "regressioneffects "
    End If
    texto = texto & ")" & CRLF$ & Tabul$
  End If
  texto = texto & "print=(all -regcmatrix -residuals "
  texto = texto & "-regressioneffects)"
  'mirar las tablas regressioneffects, averagefcsterr
  texto = texto & CRLF$'no tab
  texto = texto & "}" & CRLF$ & CRLF$'no tab

  estimate_text = texto

End Function

Function forecast_text$ ()
Dim texto$, i%, a$

    texto = ""
    texto = texto & "forecast{" & CRLF$ & Tabul$
    texto = texto & "maxlead=" & (getvar("forecast", "textforecast0"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "maxback=" & (getvar("forecast", "textforecast1"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "probability=" & (getvar("forecast", "textforecast3"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "exclude=" & (getvar("forecast", "textforecast2"))
    texto = texto & CRLF$ & Tabul$
    For i% = 0 To 2
      If Val(getvar("forecast", "save" & i%)) <> 0 Then
        a = "true"
        Exit For
      End If
    Next i%
    If a = "true" Then
      texto = texto & "save=("
      If Val(getvar("forecast", "save0")) <> 0 Then
        texto = texto & "transformed "
      End If
      If Val(getvar("forecast", "save1")) <> 0 Then
        texto = texto & "variances "
      End If
      If Val(getvar("forecast", "save2")) <> 0 Then
        texto = texto & "forecasts "
      End If
      texto = texto & ")" & CRLF$ & Tabul$
    End If
    texto = texto & "print=(all)"'mirar las tablas (libreta)
    texto = texto & CRLF$'no tab
    texto = texto & "}" & CRLF$ & CRLF$'no tab

    forecast_text = texto

End Function

Function history_text$ ()
Dim texto$, i%, a$

    texto = ""
    texto = texto & "history{" & CRLF$ & Tabul$
    If (getvar("history", "start")) <> "" Then
      texto = texto & "start="
      texto = texto & (getvar("history", "start"))
      texto = texto & CRLF$ & Tabul$
    End If
    For i% = 0 To 7
      If Val(getvar("history", "checkestimates" & i%)) <> 0 Then
        a$ = "true"
        Exit For
```

```
    End If
  Next i%
  If a$ = "true" Then
    texto = texto & "estimates=( "
    If Val(getvar("history", "checkestimates0")) <> 0 Then
      texto = texto & "sadj "
    End If
    If Val(getvar("history", "checkestimates1")) <> 0 Then
      texto = texto & "sf "
    End If
    'If Val(getvar("history", "checkestimates2")) <> 0 Then
    '  texto = texto & "sf??? "
    'End If
    If Val(getvar("history", "checkestimates3")) <> 0 Then
      texto = texto & "chng "
    End If
    If Val(getvar("history", "checkestimates4")) <> 0 Then
      texto = texto & "trend "
    End If
    If Val(getvar("history", "checkestimates5")) <> 0 Then
      texto = texto & "trendch "
    End If
    If Val(getvar("history", "checkestimates6")) <> 0 Then
      texto = texto & "aic "
    End If
    If Val(getvar("history", "checkestimates7")) <> 0 Then
      texto = texto & "fcst "
    End If
    texto = texto & ")" & CRLF$ & Tabul$
  End If
  texto = texto & "type="
  Select Case Val(getvar("history", "combotype"))
    Case 0
      texto = texto & "period"
    Case 1
      texto = texto & "all"
  End Select
  texto = texto & CRLF$ & Tabul$
  If (getvar("history", "endtable")) <> "" Then
    texto = texto & "endtable="
    texto = texto & (getvar("history", "endtable"))
    texto = texto & CRLF$ & Tabul$
  End If
  texto = texto & "revisionlag="
  texto = texto & (getvar("history", "revisionlag"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "fstep=( "
  For i% = 0 To 3
    If (getvar("history", "fstep" & i%)) <> "" Then
      texto = texto & (getvar("history", "fstep" & i%)) & " "
    End If
  Next i%
  texto = texto & ")"
  texto = texto & CRLF$ & Tabul$
  If Val(getvar("history", "checkhistory0")) <> 0 Then
    texto = texto & "fixmdl=no" & CRLF$ & Tabul$
    Else
      texto = texto & "fixmdl=yes" & CRLF$ & Tabul$
  End If
  If Val(getvar("history", "checkhistory1")) <> 0 Then
    texto = texto & "removels=yes" & CRLF$ & Tabul$
    Else
      texto = texto & "removels=no" & CRLF$ & Tabul$
  End If
  If Val(getvar("history", "checkhistory2")) <> 0 Then
    texto = texto & "fixmissing=no" & CRLF$ & Tabul$
    Else
      texto = texto & "fixmissing=yes" & CRLF$ & Tabul$
  End If
  Select Case Val(getvar("history", "refresh"))
    Case 0
      texto = texto & "refresh=yes" & CRLF$ & Tabul$
    Case 1
```

```
      texto = texto & "refresh=no" & CRLF$ & Tabul$
  End Select
  Select Case Val(getvar("history", "adjfcst"))
    Case 0
      texto = texto & "adjfcst=yes" & CRLF$ & Tabul$
    Case 1
      texto = texto & "adjfcst=no" & CRLF$ & Tabul$
  End Select
  Select Case Val(getvar("history", "outlier"))
    Case 0 'keep
      texto = texto & "outlier=keep" & CRLF$ & Tabul$
    Case 1 'remove
      texto = texto & "outlier=remove" & CRLF$ & Tabul$
    Case 2 'auto
      texto = texto & "outlier=auto" & CRLF$ & Tabul$
  End Select
  texto = texto & "outlierwin=" & (getvar("history", "outlierwin"))
  texto = texto & CRLF$ & Tabul$
  For i% = 0 To 15
    If Val(getvar("history", "save" & i%)) <> 0 Then
      a$ = "true"
      Exit For
    End If
  Next i%
  If a$ = "true" Then
    texto = texto & "save=( "
    If Val(getvar("history", "save0")) <> 0 Then
      texto = texto & "rot "
    End If
    If Val(getvar("history", "save1")) <> 0 Then
      texto = texto & "sfh "
    End If
    If Val(getvar("history", "save2")) <> 0 Then
      texto = texto & "sar "
    End If
    If Val(getvar("history", "save3")) <> 0 Then
      texto = texto & "sae "
    End If
    If Val(getvar("history", "save4")) <> 0 Then
      texto = texto & "sfr "
    End If
    If Val(getvar("history", "save5")) <> 0 Then
      texto = texto & "sfe "
    End If
    If Val(getvar("history", "save6")) <> 0 Then
      texto = texto & "chr "
    End If
    If Val(getvar("history", "save7")) <> 0 Then
      texto = texto & "che "
    End If
    If Val(getvar("history", "save8")) <> 0 Then
      texto = texto & "trr "
    End If
    If Val(getvar("history", "save9")) <> 0 Then
      texto = texto & "tre "
    End If
    If Val(getvar("history", "save10")) <> 0 Then
      texto = texto & "lkh "
    End If
    If Val(getvar("history", "save11")) <> 0 Then
      texto = texto & "fce "
    End If
    If Val(getvar("history", "save12")) <> 0 Then
      texto = texto & "tcr "
    End If
    If Val(getvar("history", "save13")) <> 0 Then
      texto = texto & "tce "
    End If
    If Val(getvar("history", "save14")) <> 0 Then
      texto = texto & "psr "
    End If
    If Val(getvar("history", "save15")) <> 0 Then
      texto = texto & "pse "
```

```
    End If
    texto = texto & ")" & CRLF$'no tab
  End If

  texto = texto & "}" & CRLF$ & CRLF$

  history_text = texto

End Function

Function identify_text$ ()
Dim texto$

 texto = ""
 texto = texto & "identify{" & CRLF$ & Tabul$
 texto = texto & "diff=" & (getvar("identify", "text0"))
 texto = texto & CRLF$ & Tabul$
 texto = texto & "sdiff=" & (getvar("identify", "textI"))
 texto = texto & CRLF$ & Tabul$
 texto = texto & "maxlag=" & (getvar("identify", "text2"))
 texto = texto & CRLF$ & Tabul$
 texto = texto & "print=(acfplot, pacfplot)"
 texto = texto & CRLF$ & "}" 'no tab
 texto = texto & CRLF$ & CRLF$ 'no tab

 identify_text = texto

End Function

Function outlier_Text$ ()
Dim texto$

 texto = ""
 texto = texto & "outlier{" & CRLF$ & Tabul$
 Select Case Val(getvar("outlier", "types"))
  Case 0
   texto = texto & "types=ao"
   texto = texto & CRLF$ & Tabul$
  Case 1
   texto = texto & "types=ls"
   texto = texto & CRLF$ & Tabul$
  Case 2
   texto = texto & "types=all"
   texto = texto & CRLF$ & Tabul$
 End Select
 texto = texto & "critical=("
 If Val(getvar("outlier", "types")) = 0 Then
  texto = texto & (getvar("outlier", "critical0"))
 End If
 If Val(getvar("outlier", "types")) = 1 Then
  texto = texto & (getvar("outlier", "critical1"))
 End If
 If Val(getvar("outlier", "types")) = 2 Then
  texto = texto & (getvar("outlier", "critical0")) & ","
  texto = texto & (getvar("outlier", "critical1"))
 End If
 texto = texto & ")" & CRLF$ & Tabul$
 texto = texto & "method="
 Select Case Val(getvar("outlier", "method"))
  Case 0
   texto = texto & "addone"
  Case 1
   texto = texto & "addall"
 End Select
 texto = texto & CRLF$ & Tabul$
 texto = texto & "span=(" & (getvar("outlier", "span0")) & ","
 texto = texto & (getvar("outlier", "span1")) & ")"
 texto = texto & CRLF$ & Tabul$
 texto = texto & "lsrun=" & (getvar("outlier", "textlsrun"))
 texto = texto & CRLF$ & Tabul$
 If Val(getvar("outlier", "iterations")) <> 0 Then
  texto = texto & "save=iterations" & CRLF$ & Tabul$
 End If
```

```
  texto = texto & "print=(all -tests)" & CRLF$ & Tabul$
  texto = texto & CRLF$ 'no tab
  texto = texto & "}" & CRLF$ & CRLF$ 'no tab

  outlier_Text = texto

End Function

Function regadjust_Text$ ()
Dim texto$, i%, a$

   texto = ""
   texto = texto & "regadjust{" & CRLF$ & Tabul$
   If Val(getvar("regadjust", "checkpriorvar")) <> 0 Then
      texto = texto & "prior=("
      If Val(getvar("regadjust", "checkprior0")) <> 0 Then
         texto = texto & "td "
      End If
      If Val(getvar("regadjust", "checkprior1")) <> 0 Then
         texto = texto & "td6 "
      End If
      If Val(getvar("regadjust", "checkprior2")) <> 0 Then
         texto = texto & "tdstock "
      End If
      If Val(getvar("regadjust", "checkprior3")) <> 0 Then
         texto = texto & "easter "
      End If
      If Val(getvar("regadjust", "checkprior4")) <> 0 Then
         texto = texto & "labor "
      End If
      If Val(getvar("regadjust", "checkprior5")) <> 0 Then
         texto = texto & "thanks "
      End If
      If Val(getvar("regadjust", "checkprior6")) <> 0 Then
         texto = texto & "ao " 'mirar
      End If
      If Val(getvar("regadjust", "checkprior7")) <> 0 Then
         texto = texto & "ls " 'mirar
      End If
      If Val(getvar("regadjust", "checkprior8")) <> 0 Then
         texto = texto & "rp "'mirar
      End If
      If Val(getvar("regadjust", "checkprior9")) <> 0 Then
         texto = texto & "user "
      End If
      texto = texto & ")" & CRLF$ & Tabul$
   End If
   If Val(getvar("regadjust", "checkfinalvar")) <> 0 Then
      texto = texto & "final=("
      If Val(getvar("regadjust", "checkfinal0")) <> 0 Then
         texto = texto & "easter "
      End If
      If Val(getvar("regadjust", "checkfinal1")) <> 0 Then
         texto = texto & "labor "
      End If
      If Val(getvar("regadjust", "checkfinal2")) <> 0 Then
         texto = texto & "thanks "
      End If
      If Val(getvar("regadjust", "checkfinal3")) <> 0 Then
         texto = texto & "ao "'mirar
      End If
      If Val(getvar("regadjust", "checkfinal4")) <> 0 Then
         texto = texto & "ls "'mirar
      End If
      If Val(getvar("regadjust", "checkfinal5")) <> 0 Then
         texto = texto & "rp "'mirar
      End If
      If Val(getvar("regadjust", "checkfinal6")) <> 0 Then
         texto = texto & "user "
      End If
      texto = texto & ")" & CRLF$ & Tabul$
   End If
   'userreg
```

```
  If Val(getvar("regadjust", "checkuser")) <> 0 Then
    texto = texto & "userreg=("
    If Val(getvar("regadjust", "optionuser0")) <> 0 Then
      texto = texto & (getvar("regadjust", "label8"))
    End If
    If Val(getvar("regadjust", "optionuser1")) <> 0 Then
    For i% = 0 To Val(getvar("regadjust", "maxuser")) - 1
      texto = texto & (getvar("regadjust", "listdifferent" & i%)) & " "
    Next i%
    End If
    texto = texto & ")" & CRLF$ & Tabul$
  End If
  texto = texto & "augmentusertd="
  Select Case Val(getvar("regadjust", "augmentusertd"))
    Case 0
      texto = texto & "yes"
    Case 1
      texto = texto & "no"
  End Select
  texto = texto & CRLF$ & Tabul$
  For i% = 0 To 5
    If Val(getvar("regadjust", "save" & i%)) <> 0 Then
      a = "true"
      Exit For
    End If
  Next i%
  If a = "true" Then
    texto = texto & "save=("
    If Val(getvar("regadjust", "save0")) <> 0 Then
      texto = texto & "aoutlier "
    End If
    If Val(getvar("regadjust", "save1")) <> 0 Then
      texto = texto & "levelshift "
    End If
    If Val(getvar("regadjust", "save2")) <> 0 Then
      texto = texto & "tradingday "
    End If
    If Val(getvar("regadjust", "save3")) <> 0 Then
      texto = texto & "holiday "
    End If
    If Val(getvar("regadjust", "save4")) <> 0 Then
      texto = texto & "userdef "
    End If
    If Val(getvar("regadjust", "save5")) <> 0 Then
      texto = texto & "outlieradjori "
    End If
    texto = texto & ")" & CRLF$'no tab
  End If
  texto = texto & "}" & CRLF$ & CRLF$

  regadjust_Text = texto

End Function

Function regression_Text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "regression{" & CRLF$ & Tabul$
  For i% = 0 To 1
    If Val(getvar("regression", "checkframevar" & i%)) <> 0 Then
      a = "true"
      Exit For
    End If
  Next i%
  If a = "true" Then
    texto = texto & "variables=("
    If Val(getvar("regression", "checkframevar0")) <> 0 Then
      If Val(getvar("regression", "checkvar0")) <> 0 Then
        texto = texto & "const "
      End If
      If Val(getvar("regression", "checkvar1")) <> 0 Then
        texto = texto & "seasonal "
```

```
    End If
    If Val(getvar("regression", "checkvar2")) <> 0 Then
      texto = texto & "sincos "'poner el valor
    End If
    If Val(getvar("regression", "checkvar3")) <> 0 Then
      texto = texto & "td "
    End If
    If Val(getvar("regression", "checkvar4")) <> 0 Then
      texto = texto & "td6 "
    End If
    If Val(getvar("regression", "checkvar5")) <> 0 Then
      texto = texto & "lom "
    End If
    If Val(getvar("regression", "checkvar6")) <> 0 Then
      texto = texto & "loq "
    End If
  End If
  If Val(getvar("regression", "checkframevar1")) <> 0 Then
    If Val(getvar("regression", "checkvar7")) <> 0 Then
      texto = texto & "tdstock[" & (getvar("regression", "tdstock")) & "] "
    End If
    If Val(getvar("regression", "checkvar8")) <> 0 Then
      texto = texto & "easter[" & (getvar("regression", "easter")) & "] "
    End If
    If Val(getvar("regression", "checkvar9")) <> 0 Then
      texto = texto & "labor[" & (getvar("regression", "labor")) & "] "
    End If
    If Val(getvar("regression", "checkvar10")) <> 0 Then
      texto = texto & "thank[" & (getvar("regression", "thank")) & "] "
    End If
    If Val(getvar("regression", "checkvar11")) <> 0 Then
    For i% = 0 To Val(getvar("regression", "maxao"))
       texto = texto & "ao" & (getvar("regression", "ao" & i%)) & " "
      Next i%
    End If
    If Val(getvar("regression", "checkvar12")) <> 0 Then
    For i% = 0 To Val(getvar("regression", "maxls"))
      texto = texto & "ls" & (getvar("regression", "ls" & i%)) & " "
    Next i%
    End If
    If Val(getvar("regression", "checkvar13")) <> 0 Then
      For i% = 0 To Val(getvar("regression", "maxrp"))
        texto = texto & "rp" & (getvar("regression", "rp" & i%)) & " "
      Next i%
    End If
    If Val(getvar("regression", "checkvar14")) <> 0 Then
      texto = texto & "lpyear "
    End If
  End If
  texto = texto & ")" & CRLF$ & Tabul$
End If
If Val(getvar("regression", "checkuservar")) <> 0 Then
  texto = texto & "user=("
  For i% = 0 To Val(getvar("regression", "maxuser"))
    texto = texto & (getvar("regression", "listuser" & i%)) & " "
  Next i%
  texto = texto & ")" & CRLF$ & Tabul$
  texto = texto & "start=" & (getvar("regression", "start"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "file='" & (getvar("regression", "file")) & "'"
  texto = texto & CRLF$ & Tabul$
  Select Case Val(getvar("regression", "comboformat"))
    Case 1
      texto = texto & "format=("
      texto = texto & (getvar("regression", "textformat")) & ")"
      texto = texto & CRLF$ & Tabul$
    Case 3
      texto = texto & "format=("
      texto = texto & "x12save" & ")"
      texto = texto & CRLF$ & Tabul$
    Case 2
      texto = texto & "format=("
      texto = texto & "edit" & ")"
```

```
        texto = texto & CRLF$ & Tabul$
    End Select
  End If
  If Val(getvar("regression", "checkaictest")) <> 0 Then
    texto = texto & "aictest="
    Select Case Val(getvar("regression", "comboaictest"))
      Case 0
        texto = texto & "td"
      Case 1
        texto = texto & "td6"
      Case 2
        If Val(getvar("regression", "checkvar7")) <> 0 Then
          texto = texto & "tdstock[" & (getvar("regression", "tdstock")) & "] "
        End If
    End Select
    texto = texto & CRLF$ & Tabul$

  'quitar la línea entera
  texto = texto & "print=(all)" & CRLF$ 'aictd

  End If
  If Val(getvar("regression", "regressionmatrix")) <> 0 Then
    texto = texto & "save=regressionmatrix" & CRLF$ 'no tab
  End If
  texto = texto & "}" & CRLF$ & CRLF$'no tab

  regression_Text = texto

End Function

Function series_text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "series{" & CRLF$ & Tabul$
  texto = texto & "start=" & getvar("series", "start")
  texto = texto & CRLF$ & Tabul$
  If Val(getvar("series", "checkspan")) <> 0 Then
     texto = texto & "span=(" & (getvar("series", "textspan0")) & "," & (getvar("series", "textspan1")) & ")"
     texto = texto & CRLF$ & Tabul$
  End If
  texto = texto & "period="
  Select Case Val(getvar("series", "period"))
    Case 0
      texto = texto & "4"
    Case 1
      texto = texto & "12"
  End Select
  texto = texto & CRLF$ & Tabul$
  If (getvar("series", "namew")) <> "" Then
     texto = texto & "name='"
     texto = texto & (getvar("series", "namew")) & "'"
     texto = texto & CRLF$ & Tabul$
  End If
  If (getvar("series", "file")) <> "" Then
     texto = texto & "file='" & (getvar("series", "file")) & "'"
     texto = texto & CRLF$ & Tabul$
  End If
  Select Case Val(getvar("series", "comboformat"))
     Case 1
        texto = texto & "format=("
        texto = texto & (getvar("series", "textformat")) & ")"
        texto = texto & CRLF$ & Tabul$
     Case 2
        texto = texto & "format=(1r)"
        texto = texto & CRLF$ & Tabul$
     Case 3
        texto = texto & "format=(2r)"
        texto = texto & CRLF$ & Tabul$
     Case 4
        texto = texto & "format=(1l)"
        texto = texto & CRLF$ & Tabul$
     Case 5
```

```
      texto = texto & "format=(2l)"
      texto = texto & CRLF$ & Tabul$
   Case 6
      texto = texto & "format=(cs)"
      texto = texto & CRLF$ & Tabul$
   Case 7
      texto = texto & "format=("
      texto = texto & "edit" & ")"
      texto = texto & CRLF$ & Tabul$
   Case 8
      texto = texto & "format=(x12save)"
      texto = texto & CRLF$ & Tabul$
End Select
'mirar el name con lo de title$
texto = texto & "decimals=" & (getvar("series", "textseries1"))
texto = texto & CRLF$ & Tabul$
texto = texto & "precision=" & (getvar("series", "textSeries0"))
texto = texto & CRLF$ & Tabul$
If Val(getvar("series", "composite")) <> 0 Then
   texto = texto & "comptype="
   Select Case (getvar("series", "comptype"))
      Case 0
         texto = texto & "add"
      Case 1
         texto = texto & "mult"
      Case 2
         texto = texto & "sub"
      Case 3
         texto = texto & "div"
   End Select
   texto = texto & CRLF$ & Tabul$
   texto = texto & "compwt=" & (getvar("series", "Compwt"))
   texto = texto & CRLF$ & Tabul$
End If
If Val(getvar("series", "Checkmodelspan")) <> 0 Then
   texto = texto & "modelspan=(" & (getvar("series", "textmodelspan0")) & "," & (getvar("series", "textmodelspan1")) & ")"
   texto = texto & CRLF$ & Tabul$
End If
If (getvar("series", "Spectrumstart")) <> "" Then
  texto = texto & "spectrumstart=" & (getvar("series", "Spectrumstart"))
  texto = texto & CRLF$ & Tabul$
End If
texto = texto & "spectrumtype="
Select Case Val(getvar("series", "spectrumtype"))
   Case 0
      texto = texto & "periodogram"
   Case 1
      texto = texto & "arspec"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "diffspectrum="
Select Case Val(getvar("series", "diffspectrum"))
  Case Is <> 0
    texto = texto & "yes"
  Case 0
    texto = texto & "no"
End Select
texto = texto & CRLF$ & Tabul$
texto = texto & "missingcode=" & (getvar("series", "missingcode"))
texto = texto & CRLF$ & Tabul$
texto = texto & "missingval=" & (getvar("series", "missingval"))
texto = texto & CRLF$ & Tabul$
texto = texto & "saveprecision=" & (getvar("series", "textseries2"))
texto = texto & CRLF$ & Tabul$
For i = 0 To 1
   If Val(getvar("series", "save" & i%)) <> 0 Then 'mirar!!!
      a = "True"
      Exit For
   End If
Next i%
If a = "True" Then
   texto = texto & "save=("
   If Val(getvar("series", "save0")) <> 0 Then
```

```
      texto = texto & "span "
    End If
    If Val(getvar("series", "save1")) <> 0 Then
      texto = texto & "missingvaladj"
    End If
    texto = texto & ")" & CRLF$ & CRLF$ & Tabul$
  End If
  texto = texto & "print=header"
  texto = texto & CRLF$ 'no tab
  texto = texto & "}" & CRLF$ & CRLF$ 'no tab

  series_text = texto

End Function

Function slidingspans_text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "slidingspans{"
  texto = texto & CRLF$ & Tabul$
  If (getvar("sliding", "start")) <> "" Then
    texto = texto & "start=" & (getvar("sliding", "start"))
    texto = texto & CRLF$ & Tabul$
  End If
  If (getvar("sliding", "length")) <> "" Then
    texto = texto & "length=" & (getvar("sliding", "length"))
    texto = texto & CRLF$ & Tabul$
  End If
  texto = texto & "cutseas=" & (getvar("sliding", "cutseas"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "cutchng=" & (getvar("sliding", "cutchng"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "cuttd=" & (getvar("sliding", "cuttd"))
  texto = texto & CRLF$ & Tabul$
  texto = texto & "fixmissing="
  Select Case Val(getvar("sliding", "fixmissing"))
    Case 0
      texto = texto & "no"
    Case 1
      texto = texto & "yes"
  End Select
  texto = texto & CRLF$ & Tabul$
  Select Case Val(getvar("sliding", "outlier"))
    Case 1
      texto = texto & "outlier=yes"
      texto = texto & CRLF$ & Tabul$
    Case 2
      texto = texto & "outlier=keep"
      texto = texto & CRLF$ & Tabul$
  End Select
  texto = texto & "fixmdl="
  Select Case Val(getvar("sliding", "fixmdl"))
    Case 0
      texto = texto & "yes"
    Case 1
      texto = texto & "no"
    Case 2
      texto = texto & "clear"
  End Select
  texto = texto & CRLF$ & Tabul$
  For i% = 0 To 8
    If Val(getvar("sliding", "save" & i%)) <> 0 Then
      a$ = "true"
      Exit For
    End If
  Next i%
  If a$ = "true" Then
    texto = texto & "save=("
    If Val(getvar("sliding", "save0")) <> 0 Then
      texto = texto & "sfspan "
    End If
    If Val(getvar("sliding", "save1")) <> 0 Then
```

```
      texto = texto & "chspan "
    End If
    If Val(getvar("sliding", "save2")) <> 0 Then
      texto = texto & "saspan "
    End If
    If Val(getvar("sliding", "save3")) <> 0 Then
      texto = texto & "ycspan "
    End If
    If Val(getvar("sliding", "save4")) <> 0 Then
      texto = texto & "tdspan "
    End If
    If Val(getvar("sliding", "save5")) <> 0 Then
      texto = texto & "sfspanind "
    End If
    If Val(getvar("sliding", "save6")) <> 0 Then
      texto = texto & "chspanind "
    End If
    If Val(getvar("sliding", "save7")) <> 0 Then
      texto = texto & "saspanind "
    End If
    If Val(getvar("sliding", "save8")) <> 0 Then
      texto = texto & "ycspanind "
    End If
    texto = texto & ")" & CRLF$ 'no Tabul$
  End If
  texto = texto & "}" & CRLF$ & CRLF$

  slidingspans_text = texto

End Function

Function transform_text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "transform{" & CRLF$ & Tabul$
  If Val(getvar("transform", "checktransform1")) <> 0 Then
    texto = texto & "function="
    Select Case Val(getvar("transform", "function"))
     Case 0
      texto = texto & "none"
     Case 1
      texto = texto & "log"
     Case 2
      texto = texto & "sqrt"
     Case 3
      texto = texto & "inverse"
     Case 4
      texto = texto & "logistic"
    End Select
    texto = texto & CRLF$ & Tabul$
  End If
  If Val(getvar("transform", "checktransform2")) <> 0 Then
    texto = texto & "power=" & (getvar("transform", "power"))
    texto = texto & CRLF$ & Tabul$
  End If
  If Val(getvar("transform", "checktransform3")) <> 0 Then
    texto = texto & "adjust="
    Select Case Val(getvar("transform", "adjust"))
      Case 0
        texto = texto & "lom"
      Case 1
        texto = texto & "loq"
      Case 2
        texto = texto & "none"
    End Select
    texto = texto & CRLF$ & Tabul$
  End If
  If Val(getvar("transform", "checktransform0")) <> 0 Then
    texto = texto & "start=" & (getvar("transform", "start"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "file="
    texto = texto & (getvar("transform", "file")) & ""
```

```
    texto = texto & CRLF$ & Tabul$
    Select Case Val(getvar("transform", "comboformat"))
      Case 1
        texto = texto & "format=("
        texto = texto & (getvar("transform", "texttransform")) & ")"
        texto = texto & CRLF$ & Tabul$
      Case 2, 3, 4, 5, 6, 8
        texto = texto & "format=("
        texto = texto & Val(getvar("transform", "comboformat" & Val(getvar("transform", "comboformat")))) & ")"
        texto = texto & CRLF$ & Tabul$
      Case 7
        texto = texto & "format=("
        texto = texto & "edit" & ")"
        texto = texto & CRLF$ & Tabul$
    End Select
    If (getvar("transform", "namew")) <> "" Then
     texto = texto & "name=" & (getvar("transform", "namew"))
     texto = texto & CRLF$ & Tabul$
    End If
    texto = texto & "precision=" & (getvar("transform", "textprecision"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "mode="
    Select Case Val(getvar("transform", "mode"))
      Case 0
        texto = texto & "percent"
      Case 1
        texto = texto & "ratio"
      Case 2
        texto = texto & "diff"
    End Select
    texto = texto & CRLF$ & Tabul$
    texto = texto & "apply="
    Select Case Val(getvar("transform", "apply"))
      Case 0
        texto = texto & "model"
      Case 1
        texto = texto & "sadj"
      Case 2
        texto = texto & "both"
    End Select
    texto = texto & CRLF$ & Tabul$
  End If
  If Val(getvar("transform", "checktransform4")) <> 0 Then
    texto = texto & "trendadjma=" & (getvar("transform", "trendadjma"))
    texto = texto & CRLF$ & Tabul$
  End If
  For i% = 0 To 3
    If Val(getvar("transform", "save" & i%)) <> 0 Then
      a = "true"
      Exit For
    End If
  Next i%
  If a = "true" Then
    texto = texto & "save=("
    If Val(getvar("transform", "save0")) <> 0 Then
      texto = texto & "priorfactors "
    End If
    If Val(getvar("transform", "save1")) <> 0 Then
      texto = texto & "prioradjusted "
    End If
    If Val(getvar("transform", "save2")) <> 0 Then
      texto = texto & "transformed "
    End If
    If Val(getvar("transform", "save3")) <> 0 Then
      texto = texto & "prioradjtrend "
    End If
    texto = texto & ")" & CRLF$ 'no tab
  End If
  texto = texto & "}" & CRLF$ & CRLF$ 'no tab

  transform_text = texto

End Function
```

```
Function x11_text$ ()
Dim texto$, i%, a$

  texto = ""
  texto = texto & "x11{" & CRLF$ & Tabul$
  texto = texto & "mode="
  Select Case Val(getvar("x11", "mode"))
    Case 0
      texto = texto & "mult"
    Case 1
      texto = texto & "add"
    Case 2
      texto = texto & "pseudoadd"
    Case 3
      texto = texto & "logadd"
  End Select
  texto = texto & CRLF$ & Tabul$
  texto = texto & "seasonalma="
  If Val(getvar("x11", "seasonalma0")) <> 0 Then
    texto = texto & "msr"
  End If
  If Val(getvar("x11", "seasonalma1")) <> 0 Then 'n
    Select Case periodicidad$
      Case "4"
        texto = texto & "("
        For i% = 12 To 15
          texto = texto & "s" & (getvar("x11", "textseasonalma" & i%)) & " "
        Next i%
        texto = texto & ")"
      Case "12"
        texto = texto & "("
        For i% = 0 To 11
          texto = texto & "s" & (getvar("x11", "textseasonalma" & i%)) & " "
        Next i%
        texto = texto & ")"
    End Select
  End If
  If Val(getvar("x11", "seasonalma2")) <> 0 Then
    Select Case Val(getvar("x11", "seasonalma"))
    Case 0
        texto = texto & "s3x3"
      Case 1
        texto = texto & "s3x5"
      Case 2
        texto = texto & "s3x9"
      Case 3
        texto = texto & "s3x15"
      Case 4
        texto = texto & "stable"
      Case 5
        texto = texto & "x11default"
    End Select
  End If
  texto = texto & CRLF$ & Tabul$
  If Val(getvar("x11", "checktrendma")) <> 0 Then
    texto = texto & "trendma="
    texto = texto & (getvar("x11", "texttrendma"))
    texto = texto & CRLF$ & Tabul$
  End If
  texto = texto & "sigmalim=("
  texto = texto & (getvar("x11", "sigmalim0")) & "," & (getvar("x11", "sigmalim1")) & ")"
  texto = texto & CRLF$ & Tabul$
  If (getvar("x11", "title")) <> "" Then
   texto = texto & "title='" & (getvar("x11", "title")) & "'"
   texto = texto & CRLF$ & Tabul$
  End If
  If Val(getvar("x11", "checkx11")) <> 0 Then
    texto = texto & "td="
    Select Case Val(getvar("x11", "combox11"))
      Case 0
        texto = texto & "apply"
      Case 1
```

```
        texto = texto & "sigf"
      Case 2
        texto = texto & "noapply"
      Case 3
        texto = texto & "prior"
      Case 4
        texto = texto & "priorsigf"
    End Select
    texto = texto & CRLF$ & Tabul$
    If (getvar("x11", "tdstart")) <> "" Then
     texto = texto & "tdstart=" & (getvar("x11", "tdstart"))
     texto = texto & CRLF$ & Tabul$
    End If
    If (getvar("x11", "tdapply")) <> "" Then
     texto = texto & "tdapply=" & (getvar("x11", "tdapply"))
     texto = texto & CRLF$ & Tabul$
    End If
    texto = texto & "tdexclude=" & (getvar("x11", "tdexclude"))
    texto = texto & CRLF$ & Tabul$
    texto = texto & "tdprior=("
    texto = texto & (getvar("x11", "tdjprior0")) & " "
    texto = texto & (getvar("x11", "tdjprior1")) & " "
    texto = texto & (getvar("x11", "tdjprior2")) & " "
    texto = texto & (getvar("x11", "tdjprior3")) & " "
    texto = texto & (getvar("x11", "tdjprior4")) & " "
    texto = texto & (getvar("x11", "tdjprior5")) & " "
    texto = texto & (getvar("x11", "tdjprior6"))
    texto = texto & ")" & CRLF$ & Tabul$
    If Val(getvar("x11", "checklom")) <> 0 Then
      texto = texto & "lom="
      Select Case Val(getvar("x11", "combolom"))
        Case 0
          texto = texto & "td"
        Case 1
          texto = texto & "seas"
      End Select
      texto = texto & CRLF$ & Tabul$
    End If
End If
'mirar lo de trading day a ver si al poner start td
'y no poner trading day adjustment da error o no hace
'nada. Mirarlo con el x12, ejecutándolo
For i% = 0 To 4
    If Val(getvar("x11", "x11holiday" & i%)) <> 0 Then
      a = "true"
      Exit For
    End If
Next i%
If a = "true" Then
    texto = texto & "x11holiday=("
    If Val(getvar("x11", "x11holiday0")) <> 0 Then
      texto = texto & "easter "
    End If
    If Val(getvar("x11", "x11holiday1")) <> 0 Then
      texto = texto & "labor "
    End If
    If Val(getvar("x11", "x11holiday2")) <> 0 Then
      texto = texto & "laborsigf "
    End If
    If Val(getvar("x11", "x11holiday3")) <> 0 Then
      texto = texto & "thank "
    End If
    If Val(getvar("x11", "x11holiday4")) <> 0 Then
      texto = texto & "thanksigf "
    End If
    texto = texto & ")" & CRLF$ & Tabul$
End If
Select Case Val(getvar("x11", "final"))
    Case 0
      texto = texto & "final=prior" & CRLF$ & Tabul$
    Case 1
      texto = texto & "final=holiday" & CRLF$ & Tabul$
End Select
```

```
'force
'forcestart
If Val(getvar("x11", "checktrendic")) <> 0 Then
   texto = texto & "trendic=" & (getvar("x11", "texttrendic"))
   texto = texto & CRLF$ & Tabul$
End If
texto = texto & "itrendma=" & (getvar("x11", "itrendma"))
texto = texto & CRLF$ & Tabul$
Select Case Val(getvar("x11", "extremeadj"))
 Case 1
   texto = texto & "extremeadj=wmad" & CRLF$ & Tabul$
 Case 2
   texto = texto & "extremeadj=wmadlog" & CRLF$ & Tabul$
End Select
Select Case Val(getvar("x11", "calendarsigma"))
 Case 0 'default
 Case 1
   texto = texto & "calendarsigma=all" & CRLF$ & Tabul$
 Case 2
   texto = texto & "calendarsigma=signif" & CRLF$ & Tabul$
 Case 3
   texto = texto & "calendarsigma=select" & CRLF$ & Tabul$
   If (getvar("x11", "sigmavec")) <> "" Then
     texto = texto & "sigmavec=(" & (getvar("x11", "sigmavec")) & ")"
     texto = texto & CRLF$ & Tabul$
   End If
End Select
texto = texto & "type="
Select Case Val(getvar("x11", "type"))
 Case Is <> 0
   texto = texto & "sa"
 Case 0
   texto = texto & "sumry"
End Select
texto = texto & CRLF$ & Tabul$
If (getvar("x11", "textdmaxlead")) <> "" Then
 texto = texto & "tdmaxlead=" & (getvar("x11", "texttdmaxlead"))
 texto = texto & CRLF$ & Tabul$
End If
texto = texto & "taper=" & (getvar("x11", "taper")) & CRLF$ & Tabul$
'texto = texto & "shortsf="
'Select Case Val(getvar("x11", "shortsf"))
' Case 0
'   texto = texto & "no"
' Case 1
'   texto = texto & "yes"
'End Select
'texto = texto & CRLF$ & Tabul$
texto = texto & "strike="
Select Case Val(getvar("x11", "strike"))
   Case Is <> 0
      texto = texto & "yes"
   Case 0
      texto = texto & "no"
End Select
texto = texto & CRLF$ & Tabul$
For i% = 0 To 60
   If Val(getvar("x11", "checkx11" & i%)) <> 0 Then
      a = "true"
      Exit For
   End If
Next i%
If a = "true" Then
   texto = texto & "save=("
   If Val(getvar("x11", "checkx110")) <> 0 Then
      texto = texto & "h1 "
   End If
   If Val(getvar("x11", "checkx111")) <> 0 Then
      texto = texto & "a4 "
   End If
   If Val(getvar("x11", "checkx112")) <> 0 Then
      texto = texto & "b1 "
   End If
```

```
If Val(getvar("x11", "checkx113")) <> 0 Then
  texto = texto & "c16 "
End If
If Val(getvar("x11", "checkx114")) <> 0 Then
   texto = texto & "c17 "
End If
If Val(getvar("x11", "checkx115")) <> 0 Then
   texto = texto & "c18 "
End If
If Val(getvar("x11", "checkx116")) <> 0 Then
   texto = texto & "d8 "
End If
If Val(getvar("x11", "checkx117")) <> 0 Then
   texto = texto & "d9 "
End If
If Val(getvar("x11", "checkx118")) <> 0 Then
   texto = texto & "d10 "
End If
If Val(getvar("x11", "checkx119")) <> 0 Then
   texto = texto & "fsd "
End If
If Val(getvar("x11", "checkx1110")) <> 0 Then
   texto = texto & "d11 "
End If
If Val(getvar("x11", "checkx1111")) <> 0 Then
   texto = texto & "saa "
End If
If Val(getvar("x11", "checkx1112")) <> 0 Then
   texto = texto & "rnd "
End If
If Val(getvar("x11", "checkx1113")) <> 0 Then
   texto = texto & "d12 "
End If
If Val(getvar("x11", "checkx114")) <> 0 Then
   texto = texto & "d13 "
End If
If Val(getvar("x11", "checkx1115")) <> 0 Then
   texto = texto & "d16 "
End If
If Val(getvar("x11", "checkx1I16")) <> 0 Then
   texto = texto & "fad "
End If
If Val(getvar("x11", "checkx1I17")) <> 0 Then
   texto = texto & "e4 "
End If
If Val(getvar("x11", "checkx1118")) <> 0 Then
   texto = texto & "e5 "
End If
If Val(getvar("x11", "checkx1119")) <> 0 Then
   texto = texto & "e6 "
End If
If Val(getvar("x11", "checkx1120")) <> 0 Then
   texto = texto & "e6a "
End If
If Val(getvar("x11", "checkx1121")) <> 0 Then
   texto = texto & "e6r "
End If
If Val(getvar("x11", "checkx1122")) <> 0 Then
   texto = texto & "e7 "
End If
If Val(getvar("x11", "checkx1123")) <> 0 Then
   texto = texto & "b2 "
End If
If Val(getvar("x11", "checkx1124")) <> 0 Then
   texto = texto & "b3 "
End If
If Val(getvar("x11", "checkx1125")) <> 0 Then
   texto = texto & "b5 "
End If
If Val(getvar("x11", "checkx1126")) <> 0 Then
   texto = texto & "b6 "
End If
If Val(getvar("x11", "checkx1127")) <> 0 Then
```

```
    texto = texto & "b7 "
End If
If Val(getvar("x11", "checkx1128")) <> 0 Then
    texto = texto & "b8 "
End If
If Val(getvar("x11", "checkx1129")) <> 0 Then
    texto = texto & "b10 "
End If
If Val(getvar("x11", "checkx1130")) <> 0 Then
    texto = texto & "b11 "
End If
If Val(getvar("x11", "checkx1131")) <> 0 Then
    texto = texto & "b13 "
End If
If Val(getvar("x11", "checkx1132")) <> 0 Then
    texto = texto & "b16 "
End If
If Val(getvar("x11", "checkx1133")) <> 0 Then
    texto = texto & "b17 "
End If
If Val(getvar("x11", "checkx1134")) <> 0 Then
    texto = texto & "b18 "
End If
If Val(getvar("x11", "checkx1135")) <> 0 Then
    texto = texto & "b19 "
End If
If Val(getvar("x11", "checkx1136")) <> 0 Then
    texto = texto & "b20 "
End If
If Val(getvar("x11", "checkx1137")) <> 0 Then
    texto = texto & "c1 "
End If
If Val(getvar("x11", "checkx1138")) <> 0 Then
    texto = texto & "c2 "
End If
If Val(getvar("x11", "checkx1139")) <> 0 Then
    texto = texto & "c4 "
End If
If Val(getvar("x11", "checkx1140")) <> 0 Then
    texto = texto & "c5 "
End If
If Val(getvar("x11", "checkx1141")) <> 0 Then
    texto = texto & "c6 "
End If
If Val(getvar("x11", "checkx1142")) <> 0 Then
    texto = texto & "c7 "
End If
If Val(getvar("x11", "checkx1143")) <> 0 Then
    texto = texto & "c9 "
End If
If Val(getvar("x11", "checkx1144")) <> 0 Then
    texto = texto & "c10 "
End If
If Val(getvar("x11", "checkx1145")) <> 0 Then
    texto = texto & "c11 "
End If
If Val(getvar("x11", "checkx1146")) <> 0 Then
    texto = texto & "c13 "
End If
If Val(getvar("x11", "checkx1147")) <> 0 Then
    texto = texto & "c19 "
End If
If Val(getvar("x11", "checkx1148")) <> 0 Then
    texto = texto & "c20 "
End If
If Val(getvar("x11", "checkx1149")) <> 0 Then
    texto = texto & "d1 "
End If
If Val(getvar("x11", "checkx1150")) <> 0 Then
  texto = texto & "d2 "
End If
If Val(getvar("x11", "checkx1151")) <> 0 Then
    texto = texto & "d4 "
```

```
    End If
    If Val(getvar("x11", "checkx1152")) <> 0 Then
      texto = texto & "d5 "
    End If
    If Val(getvar("x11", "checkx1153")) <> 0 Then
      texto = texto & "d6 "
    End If
    If Val(getvar("x11", "checkx1154")) <> 0 Then
      texto = texto & "d7 "
    End If
    If Val(getvar("x11", "checkx1155")) <> 0 Then
      texto = texto & "tal "
    End If
    If Val(getvar("x11", "checkx1156")) <> 0 Then
      texto = texto & "iao "
    End If
    If Val(getvar("x11", "checkx1157")) <> 0 Then
      texto = texto & "e1 "
    End If
    If Val(getvar("x11", "checkx1158")) <> 0 Then
      texto = texto & "e2 "
    End If
    If Val(getvar("x11", "checkx1159")) <> 0 Then
      texto = texto & "e3 "
    End If
    If Val(getvar("x11", "checkx1160")) <> 0 Then
      texto = texto & "e11 "
    End If
    If Val(getvar("x11", "checkx1161")) <> 0 Then
      texto = texto & "f1 "
    End If
    texto = texto & ")" & CRLF$ & Tabul$
  End If
  texto = texto & "appendfcst="
  Select Case Val(getvar("x11", "appendfcst"))
   Case Is <> 0
    texto = texto & "yes"
   Case 0
    texto = texto & "no"
  End Select
  texto = texto & CRLF$ & Tabul$
  texto = texto & "print=(ftesthol, extremetd,tdreg,ftestd8,movseasrat,"
  texto = texto & "residualseasf,x11diag,qstat,tdaytype,specsa,specirr,ftestb1)"
  'texto = texto & ",replacsib4,replacsib9,extremetdb,tdregb15,autosf)"
  texto = texto & CRLF$ 'no tab
  texto = texto & "}" & CRLF$ & CRLF$ 'no tab

  x11_text = texto


End Function
```

1.2.4. Run.bas

```
Option Explicit
Global runmode%

Sub configur_0 ()
Dim x%, auxname$

   formx12.Show
   menuprincipal.Hide
   'activar series
   Call setvar("series", "cargar", "yes")
   Call setvar("formx12", "arguments0", True)
   formx12.SSIndexTabParametros.TabEnabled(0) = True
   auxname$ = menuprincipal.List.List(0)
   Call setvar("series", "file", auxname)
   auxname$ = (getvar("series", "file"))
   Call setvar("series", "name", series_name(CStr(auxname)))
   series.LabelSeriesFile.Caption = menuprincipal.List.List(0)
   series.LabelSeriesName.Caption = series_name(CStr(series.LabelSeriesFile.Caption))

End Sub

Sub configur_2 ()
Dim x%, auxfile$

   'new spec file
   X12IniFile$ = app.Path & "\output\x12conf.dat"
   If Dir$(X12IniFile$) <> "" Then
      Kill X12IniFile$
   End If

   Call titleform

   'desactivar todos los arguments
   For x% = 0 To 14
      'formx12.SSIndexTabParametros.TabEnabled(x%) = False
      Call setvar("formx12", "arguments" & x%, False)
   Next x%

   For x% = 1 To 15
      checkcargar(x%) = "no"
   Next x%
   Call setvar("series", "cargar", checkcargar(1))
   Call setvar("composite", "cargar", checkcargar(2))
   Call setvar("transform", "cargar", checkcargar(3))
   Call setvar("x11", "cargar", checkcargar(4))
   Call setvar("identify", "cargar", checkcargar(5))
   Call setvar("regression", "cargar", checkcargar(6))
   Call setvar("arima", "cargar", checkcargar(7))
   Call setvar("automdl", "cargar", checkcargar(8))
   Call setvar("estimate", "cargar", checkcargar(9))
   Call setvar("outlier", "cargar", checkcargar(10))
   Call setvar("check", "cargar", checkcargar(11))
   Call setvar("forecast", "cargar", checkcargar(12))
   Call setvar("regadjust", "cargar", checkcargar(13))
   Call setvar("sliding", "cargar", checkcargar(14))
   Call setvar("history", "cargar", checkcargar(15))

   Call inicializar_series

   Call inicializar

   Call inicializar_composite
   Call inicializar_transform
   Call inicializar_x11
   Call inicializar_identify
   Call inicializar_regression
   Call inicializar_arima
   Call inicializar_automdl
   Call inicializar_estimate
   Call inicializar_outlier
   Call inicializar_check
```

```
    Call inicializar_forecast
    Call inicializar_regadjust
    Call inicializar_sliding
    Call inicializar_history

    'fin new spec file
    formx12.Show
    menuprincipal.Hide
'   'activar series
    Call setvar("series", "cargar", "yes")
    Call setvar("formx12", "arguments0", True)
    formx12.SSIndexTabParametros.TabEnabled(0) = True
    auxfile$ = menuprincipal.List.List(menuprincipal.List.ListCount - 1)
    Call setvar("series", "file", auxfile$)
    'no se si alguna vez estará vacío, en principio nunca estará
    'vacío este apartado porque se le llama desde la lista cuando
    'añade algún elemento, por eso siempre habrá algo, pero por si acaso
    Call setvar("series", "name", series_name(CStr((getvar("series", "file")))))


End Sub

Sub configur_2a ()
Dim x%, auxfile$

    'new spec file
    X12IniFile$ = app.Path & "\output\x12conf.dat"
    If Dir$(X12IniFile$) <> "" Then
       Kill X12IniFile$
    End If

    Call titleform

    'desactivar todos los arguments
    For x% = 0 To 14
       'formx12.SSIndexTabParametros.TabEnabled(x%) = False
       Call setvar("formx12", "arguments" & x%, False)
    Next x%
    For x% = 1 To 15
       checkcargar(x%) = "no"
    Next x%
    Call setvar("series", "cargar", checkcargar(1))
    Call setvar("composite", "cargar", checkcargar(2))
    Call setvar("transform", "cargar", checkcargar(3))
    Call setvar("x11", "cargar", checkcargar(4))
    Call setvar("identify", "cargar", checkcargar(5))
    Call setvar("regression", "cargar", checkcargar(6))
    Call setvar("arima", "cargar", checkcargar(7))
    Call setvar("automdl", "cargar", checkcargar(8))
    Call setvar("estimate", "cargar", checkcargar(9))
    Call setvar("outlier", "cargar", checkcargar(10))
    Call setvar("check", "cargar", checkcargar(11))
    Call setvar("forecast", "cargar", checkcargar(12))
    Call setvar("regadjust", "cargar", checkcargar(13))
    Call setvar("sliding", "cargar", checkcargar(14))
    Call setvar("history", "cargar", checkcargar(15))

    Call inicializar_series

    Call inicializar

    Call inicializar_composite
    Call inicializar_transform
    Call inicializar_x11
    Call inicializar_identify
    Call inicializar_regression
    Call inicializar_arima
    Call inicializar_automdl
    Call inicializar_estimate
    Call inicializar_outlier
    Call inicializar_check
    Call inicializar_forecast
    Call inicializar_regadjust
```

```
    Call inicializar_sliding
    Call inicializar_history

    'fin new spec file

    'MIRAR DE HACERLO SIN QUE SE HAGA NEW SPEC FILE

    auxfile$ = ""
    For x% = 0 To menuprincipal.List.ListCount - 1
     If menuprincipal.List.Selected(x%) = True Then
      auxfile$ = menuprincipal.List.List(x%)
      Exit For
     End If
    Next x%
    If auxfile$ = "" Then
     response = MsgBox("Non of the series were selected", 48, title)
     Exit Sub
     Else
      formx12.Show
      menuprincipal.Hide
      'activar series
      Call setvar("series", "cargar", "yes")
      Call setvar("formx12", "arguments0", True)
      formx12.SSIndexTabParametros.TabEnabled(0) = True
      Call setvar("series", "file", auxfile$)
      'no se si alguna vez estará vacío, en principio nunca estará
      'vacío este apartado porque se le llama desde la lista cuando
      'añade algún elemento, por eso siempre habrá algo, pero por si acaso
      Call setvar("series", "name", series_name(CStr((getvar("series", "file")))))
    End If

End Sub

Sub configur_3 ()
Dim x%, auxfile$

    'new spec file
    X12IniFile$ = app.Path & "\output\x12conf.dat"
    If Dir$(X12IniFile$) <> "" Then
       Kill X12IniFile$
    End If

    Call titleform

    'desactivar todos los arguments
    For x% = 0 To 14
       'formx12.SSIndexTabParametros.TabEnabled(x%) = False
       Call setvar("formx12", "arguments" & x%, False)
    Next x%

    For x% = 1 To 15
       checkcargar(x%) = "no"
    Next x%
    Call setvar("series", "cargar", checkcargar(1))
    Call setvar("composite", "cargar", checkcargar(2))
    Call setvar("transform", "cargar", checkcargar(3))
    Call setvar("x11", "cargar", checkcargar(4))
    Call setvar("identify", "cargar", checkcargar(5))
    Call setvar("regression", "cargar", checkcargar(6))
    Call setvar("arima", "cargar", checkcargar(7))
    Call setvar("automdl", "cargar", checkcargar(8))
    Call setvar("estimate", "cargar", checkcargar(9))
    Call setvar("outlier", "cargar", checkcargar(10))
    Call setvar("check", "cargar", checkcargar(11))
    Call setvar("forecast", "cargar", checkcargar(12))
    Call setvar("regadjust", "cargar", checkcargar(13))
    Call setvar("sliding", "cargar", checkcargar(14))
    Call setvar("history", "cargar", checkcargar(15))

    Call inicializar_series

    Call inicializar
```

```
    Call inicializar_composite
    Call inicializar_transform
    Call inicializar_x11
    Call inicializar_identify
    Call inicializar_regression
    Call inicializar_arima
    Call inicializar_automdl
    Call inicializar_estimate
    Call inicializar_outlier
    Call inicializar_check
    Call inicializar_forecast
    Call inicializar_regadjust
    Call inicializar_sliding
    Call inicializar_history

    'fin new spec file
    formx12.Show
    menuprincipal.Hide
    'activar series
    Call setvar("series", "cargar", "yes")
    Call setvar("formx12", "arguments0", True)
    formx12.SSIndexTabParametros.TabEnabled(0) = True
    auxfile$ = menuprincipal.List.List(menuprincipal.List.ListCount - 1)
    Call setvar("series", "file", auxfile$)
    'no se si alguna vez estará vacío, en principio nunca estará
    'vacío este apartado porque se le llama desde la lista cuando
    'añade algún elemento, por eso siempre habrá algo, pero por si acaso
    Call setvar("series", "name", series_name(CStr((getvar("series", "file")))))
    Call setvar("series", "composite", True)

End Sub

Sub configur_3a ()
Dim x%

    'new spec file
    X12IniFile$ = app.Path & "\output\x12conf.dat"
    If Dir$(X12IniFile$) <> "" Then
       Kill X12IniFile$
    End If

    Call titleform

    'desactivar todos los arguments
    For x% = 0 To 14
       'formx12.SSIndexTabParametros.TabEnabled(x%) = False
       Call setvar("formx12", "arguments" & x%, False)
    Next x%

    For x% = 1 To 15
       checkcargar(x%) = "no"
    Next x%
    Call setvar("series", "cargar", checkcargar(1))
    Call setvar("composite", "cargar", checkcargar(2))
    Call setvar("transform", "cargar", checkcargar(3))
    Call setvar("x11", "cargar", checkcargar(4))
    Call setvar("identify", "cargar", checkcargar(5))
    Call setvar("regression", "cargar", checkcargar(6))
    Call setvar("arima", "cargar", checkcargar(7))
    Call setvar("automdl", "cargar", checkcargar(8))
    Call setvar("estimate", "cargar", checkcargar(9))
    Call setvar("outlier", "cargar", checkcargar(10))
    Call setvar("check", "cargar", checkcargar(11))
    Call setvar("forecast", "cargar", checkcargar(12))
    Call setvar("regadjust", "cargar", checkcargar(13))
    Call setvar("sliding", "cargar", checkcargar(14))
    Call setvar("history", "cargar", checkcargar(15))

    Call inicializar_series

    Call inicializar

    Call inicializar_composite
```

```
    Call inicializar_transform
    Call inicializar_x11
    Call inicializar_identify
    Call inicializar_regression
    Call inicializar_arima
    Call inicializar_automdl
    Call inicializar_estimate
    Call inicializar_outlier
    Call inicializar_check
    Call inicializar_forecast
    Call inicializar_regadjust
    Call inicializar_sliding
    Call inicializar_history

    'fin new spec file
    formx12.Show
    menuprincipal.Hide
    'activar composite
    Call setvar("composite", "cargar", "yes")
    Call setvar("formx12", "arguments1", True)
    formx12.SSIndexTabParametros.TabEnabled(1) = True
    Call setvar("composite", "name", menuprincipal.TextSeriesComposite)

End Sub

Sub configur_3b ()
Dim x%, auxfile$

    'new spec file
    X12IniFile$ = app.Path & "\output\x12conf.dat"
    If Dir$(X12IniFile$) <> "" Then
       Kill X12IniFile$
    End If

    Call titleform

    'desactivar todos los arguments
    For x% = 0 To 14
       'formx12.SSIndexTabParametros.TabEnabled(x%) = False
       Call setvar("formx12", "arguments" & x%, False)
    Next x%

    For x% = 1 To 15
       checkcargar(x%) = "no"
    Next x%
    Call setvar("series", "cargar", checkcargar(1))
    Call setvar("composite", "cargar", checkcargar(2))
    Call setvar("transform", "cargar", checkcargar(3))
    Call setvar("x11", "cargar", checkcargar(4))
    Call setvar("identify", "cargar", checkcargar(5))
    Call setvar("regression", "cargar", checkcargar(6))
    Call setvar("arima", "cargar", checkcargar(7))
    Call setvar("automdl", "cargar", checkcargar(8))
    Call setvar("estimate", "cargar", checkcargar(9))
    Call setvar("outlier", "cargar", checkcargar(10))
    Call setvar("check", "cargar", checkcargar(11))
    Call setvar("forecast", "cargar", checkcargar(12))
    Call setvar("regadjust", "cargar", checkcargar(13))
    Call setvar("sliding", "cargar", checkcargar(14))
    Call setvar("history", "cargar", checkcargar(15))

    Call inicializar_series

    Call inicializar

    Call inicializar_composite
    Call inicializar_transform
    Call inicializar_x11
    Call inicializar_identify
    Call inicializar_regression
    Call inicializar_arima
    Call inicializar_automdl
    Call inicializar_estimate
```

```
  Call inicializar_outlier
  Call inicializar_check
  Call inicializar_forecast
  Call inicializar_regadjust
  Call inicializar_sliding
  Call inicializar_history

  'fin new spec file

  'MIRAR DE HACERLO SIN QUE HAGA NEW SPEC FILE

  auxfile$ = ""
  For x% = 0 To menuprincipal.List.ListCount - 1
   If menuprincipal.List.Selected(x%) = True Then
    auxfile$ = menuprincipal.List.List(x%)
    Exit For
   End If
  Next x%
  If auxfile$ = "" Then
   response = MsgBox("Non of the series were selected", 48, title)
   Exit Sub
   Else
    formx12.Show
    menuprincipal.Hide
    'activar series
    Call setvar("series", "cargar", "yes")
    Call setvar("formx12", "arguments0", True)
    formx12.SSIndexTabParametros.TabEnabled(0) = True
    Call setvar("series", "file", auxfile$)
    Call setvar("series", "name", series_name(CStr((getvar("series", "file")))))
    Call setvar("series", "composite", True)
  End If

End Sub

Sub checkerror ()
Dim x%, aux$, texterror$, num%

 Select Case runmode%
  Case 0 'normal mode
   seriesname = app.Path & "\output\" & series_name(CStr((getvar("series", "file")))) & ".err"
   Open seriesname For Input As 1
   texterror$ = Input(LOF(1), 1)
   Close #1
   num% = InStr(texterror$, "ERROR:")
   If num% <> 0 Then
    Load errorform
    errorform.Show
   End If
  Case 1 'single spec mode
   texterror$ = ""
   For x% = 0 To menuprincipal.List.ListCount - 1
    aux$ = menuprincipal.List.List(x%)
    seriesname = app.Path & "\output\" & series_name(aux$) & ".err"
    If Dir$(seriesname) <> "" Then 'n
      Open seriesname For Input As 1
      texterror$ = texterror$ & Input(LOF(1), 1)
      Close #1
    End If 'n
   Next x%
   num% = InStr(texterror$, "ERROR:")
   If num% <> 0 Then
    Load errorform
    errorform.Show
   End If
  Case 2 'multi spec mode
   texterror$ = ""
   For x% = 0 To menuprincipal.List.ListCount - 1
    aux$ = menuprincipal.List.List(x%)
    seriesname = app.Path & "\output\" & series_name(aux$) & ".err"
    If Dir$(seriesname) <> "" Then 'n
      Open seriesname For Input As 1
      texterror$ = texterror$ & Input(LOF(1), 1)
```

```
        Close #1
       End If 'n
      Next x%
      num% = InStr(texterror$, "ERROR:")
      If num% <> 0 Then
       Load errorform
       errorform.Show
      End If
     Case 3 'composite series
      texterror$ = ""
      For x% = 0 To menuprincipal.List.ListCount - 1
       aux$ = menuprincipal.List.List(x%)
       seriesname = app.Path & "\output\" & series_name(aux$) & ".err"
       If Dir$(seriesname) <> "" Then 'n
          Open seriesname For Input As 1
          texterror$ = texterror$ & Input(LOF(1), 1)
          Close #1
       End If 'n
      Next x%
      aux$ = menuprincipal.TextSeriesComposite
      seriesname = app.Path & "\output\" & series_name(aux$) & ".err"
      If Dir$(seriesname) <> "" Then 'n
       Open seriesname For Input As 1
       texterror$ = texterror$ & Input(LOF(1), 1)
       Close #1
      End If 'n
      num% = InStr(texterror$, "ERROR:")
      If num% <> 0 Then
       Load errorform
       errorform.Show
      End If
    End Select

End Sub

Sub series_0 ()
Dim x%, igual$

 On Error GoTo errorseries0

   menuprincipal.CMDialogMenu.Filter = "Archivos de Datos (*.DAT) |*.dat|Todos los archivos |*.*"
   menuprincipal.CMDialogMenu.Filename = ""
   menuprincipal.CMDialogMenu.Action = 1
'  igual$ = "no"
'  If menuprincipal.CMDialogMenu.Filename <> "" Then
'    For x% = 0 To menuprincipal.List.ListCount - 1
'      If menuprincipal.CMDialogMenu.Filename = menuprincipal.List.List(x%) Then
'        igual$ = "yes"
'      End If
'    Next x%
'    If igual$ <> "yes" Then
'      menuprincipal.List.AddItem menuprincipal.CMDialogMenu.Filename
'    End If
'  End If
'  Exit Sub

   menuprincipal.List.List(0) = menuprincipal.CMDialogMenu.Filename 'n
   Exit Sub 'n

errorseries0:
  Select Case Err
   Case 32755
    Exit Sub
  End Select

End Sub

Sub series_1 ()
Dim x%, igual$

 On Error GoTo errorseries1

   menuprincipal.CMDialogMenu.Filter = "Archivos de Datos (*.DAT) |*.dat|Todos los archivos |*.*"
```

```
    menuprincipal.CMDialogMenu.Filename = ""
    menuprincipal.CMDialogMenu.Action = 1
    igual$ = "no"
    If menuprincipal.CMDialogMenu.Filename <> "" Then
     For x% = 0 To menuprincipal.List.ListCount - 1
      If menuprincipal.CMDialogMenu.Filename = menuprincipal.List.List(x%) Then
       igual$ = "yes"
      End If
     Next x%
     If igual$ <> "yes" Then
      menuprincipal.List.AddItem menuprincipal.CMDialogMenu.Filename
     End If
    End If
    Exit Sub

errorseries1:
  Select Case Err
    Case 32755
      Exit Sub
  End Select

End Sub

Sub series_2 ()
Dim x%, igual$

  On Error GoTo errorseries2

    menuprincipal.CMDialogMenu.Filter = "Archivos de Datos (*.DAT) |*.dat|Todos los archivos |*.*"
    menuprincipal.CMDialogMenu.Filename = ""
    menuprincipal.CMDialogMenu.Action = 1
    igual$ = "no"
    If menuprincipal.CMDialogMenu.Filename <> "" Then
     For x% = 0 To menuprincipal.List.ListCount - 1
      If menuprincipal.CMDialogMenu.Filename = menuprincipal.List.List(x%) Then
       igual$ = "yes"
      End If
     Next x%
     If igual$ <> "yes" Then
      menuprincipal.List.AddItem menuprincipal.CMDialogMenu.Filename
      Call configur_2
     End If
    End If
    Exit Sub

errorseries2:
  Select Case Err
    Case 32755
      Exit Sub
  End Select

End Sub

Sub series_3 ()
Dim x%, igual$

  On Error GoTo errorseries3a

    menuprincipal.CMDialogMenu.Filter = "Archivos de Datos (*.DAT) |*.dat|Todos los archivos |*.*"
    menuprincipal.CMDialogMenu.Filename = ""
    menuprincipal.CMDialogMenu.Action = 1
    igual$ = "no"
    If menuprincipal.CMDialogMenu.Filename <> "" Then
     For x% = 0 To menuprincipal.List.ListCount - 1
      If menuprincipal.CMDialogMenu.Filename = menuprincipal.List.List(x%) Then
       igual$ = "yes"
      End If
     Next x%
     If igual$ <> "yes" Then
      menuprincipal.List.AddItem menuprincipal.CMDialogMenu.Filename
      Call configur_3
     End If
    End If
```

```
    Exit Sub

errorseries3a:
  Select Case Err
    Case 32755
      Exit Sub
  End Select

End Sub

Sub configur_1 ()
Dim x%

    formx12.Show
    menuprincipal.Hide
    'activar series
    Call setvar("series", "cargar", "yes")
    Call setvar("formx12", "arguments0", True)
    'hacerlo con el call setvar
    formx12.SSIndexTabParametros.TabEnabled(0) = True
    'series.labelseriesfile.caption = ""
    'series.labelseriesname.caption = ""
    Call setvar("series", "file", "")
    series.LabelSeriesFile.Enabled = False
    series.Command3DSeriesFile.Enabled = False
    series.LabelSeriesName.Enabled = False
    Call setvar("series", "file", "")
    Call setvar("series", "name", "")
    'Call setvar("series", "Searchfile", series!Command3DSeriesFile)

End Sub
```

1.2.5. Graph.bas

```
Option Explicit
Global GraphArray!(), GraphArraylen%(), col%, datagraph$'nombre del fichero de datos
Global seriesgraph$

Function cargar_datos% ()
Dim r%, n$, fn%

 'poner en datagraph$ el fichero correspondiente
 graph_file

 fn% = FreeFile
 Open datagraph$ For Input As 1
  'Select Case Mid$(tables.CMDialog1.Filename, Len(tables.CMDialog1.Filename) - 2, 3)
  Select Case Mid$(datagraph$, Len(datagraph$) - 2, 3)
   Case ".A1", "MVA", ".A2", ".A3", "A10", "TRN", ".B1", ".B2", ".B3", ".B5"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case "B10", "B11", "B13", "B17", "B20", ".C1", ".C2", ".C4", ".C5", ".C6"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case ".C7", "C10", "C11", "C13", "C17", "C20", ".D1", ".D2", ".D4", ".D5"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case ".D6", ".D7", ".D8", ".D9", "D10", "D11", "D12", "D13", "D16", ".E1"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case ".E2", ".E5", ".E6", "E11", "FSD", "SAA"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case "ACF", "PCF", "RSD", "FTR", "FVR", "FCT", "TAL"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case ".AO", ".LS", ".TD", "HOL", "USR", "A11" 'mirar
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case "ROT", "SFH", "SAR", "SAE", "SFR", "SFE", "CHR", "CHE", "TRR", "TRE"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1
   Case "LKH", "FCE", "TCR", "TCE", "PSR", "PSE"
    r% = Cargar_datos1(fn%, col%)
    col% = col% + 1

   'Case ".H1", ".A4", "C16", "C18", ".E4"  'mirarlos
  End Select
 Close #fn%

 cargar_datos% = r% - 1

End Function

Sub graph_file () 'elegir el fichero según el gráfico

 'Select Case tables.TextGraph.Text
 Select Case tables.ListGraph.List(tables.ListGraph.ListIndex)
  Case "Time series"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".A1"
  Case "Time series with missing values replaced"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".MVA"
  Case "Prior adjustment factors"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".A2"
  Case "Prior adjusted series"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".A3"
  Case "Prior adjusted and transformed series"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".TRN"
  Case "Henderson trend of prior adjusted series"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".A10"
  Case "ACF"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".ACF"
  Case "PACF"
   datagraph$ = app.Path & "\output\" & seriesgraph & ".PCF"
  Case "Residuals"
```

```
  datagraph$ = app.Path & "\output\" & seriesgraph & ".RSD"
Case "Final seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D11"
Case "Final seasonally adjusted series with constrained yearly totals"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".SAA"
Case "Rounded final seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".RND"
Case "Percent changes in seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E6"
Case "Percent changes in seasonally adjusted series with revised yearly totals"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E6A"
Case "Percent changes in rounded seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E6R"
Case "Preliminary seasonally adjusted series, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B6"
Case "Seasonally adjusted series, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B11"
Case "Preliminary seasonally adjusted series, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C6"
Case "Seasonally adjusted series, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C11"
Case "Preliminary seasonally adjusted series, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D6"
Case "Seasonally adjusted series modified for extreme values"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E2"
Case "Robust final seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E11"
Case "MCD moving average of the final seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".F1"
Case "Final trend-cycle"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D12"
Case "Final trend-cycle using original series modified for extreme values"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E7"
Case "Preliminary trend-cycle, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B2"
Case "Preliminary trend-cycle, B iteration ??"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B7"
Case "Preliminary trend-cycle, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C2"
Case "Preliminary trend-cycle, C iteration ??"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C7"
Case "Preliminary trend-cycle, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D2"
Case "Preliminary trend-cycle, D iteration ??"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D7"
Case "Final trend-cycle adjusted for LS"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".TAL"
Case "Final weights for the irregular component"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C17"
Case "Final irregular component"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D13"
Case "Irregular component, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B13"
Case "Preliminary weights for the irregular component"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B17"
Case "Irregular component, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C13"
Case "Final irregular component adjusted for AO"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".IAO"
Case "Irregular component modified for extreme values"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E3"
Case "Final seasonal factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D10"
Case "Final seasonal difference"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".FSD"
Case "Combined seasonal and trading day factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D16"
Case "Preliminary seasonal factors, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B5"
Case "Seasonal factors, B iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B10"
Case "Preliminary seasonal factors, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C5"
```

```
Case "Preliminary seasonal factors, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D5"
Case "Prior tradind day factors and weights"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".A4"
Case "Final trading day factors and weights"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C16"
Case "Final trading day from combined daily weights"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C18"
Case "Combined seasonal and trading day factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D16"
Case "Preliminary trading day factors and weights"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B16"
Case "Preliminary trading day from combined daily weights"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B18"
Case "Original series adjusted for preliminary trading day"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B19"
Case "Combined X-11 holiday adjustment factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".H1"
Case "Final unmodified si-ratios"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D8"
Case "Final replacement values for si-ratios, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D9"
Case "Preliminary unmodified si-ratios"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B3"
Case "Unmodified si-ratios"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B8"
Case "Modified si-ratios, C iteration ??"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C4"
Case "Modified si-ratios, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C9"
Case "Modified si-ratios, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D4"
Case "Forecasts on the transformated scale"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".FTR"
Case "Forecast error variances on the transformated scale"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".FVR"
Case "Forecasts on the original scale"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".FCT"
Case "regARIMA Additive outlier factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".AO"
Case "regARIMA Level change and ramp factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".LS"
Case "regARIMA Trading day factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".TD"
Case "regARIMA Holiday factors"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".HOL"
Case "Factors from user-defined regression variables"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".USR"
Case "Original series adjusted for regARIMA outliers"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".A11"
Case "Original series, adjusted for prior effects and forecast extended"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B1"
Case "Ratio of yearly totals for original, seasonally adjusted series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E4"
Case "Percent changes in original series"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E5"
Case "Original series adjusted for preliminary trading day"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".B19"
Case "Original series modified for outliers, td and prior factors, C iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C1"
Case "Original series adjusted for final trading day"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".C19"
Case "Original series modified for outliers, td and prior factors, D iteration"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".D1"
Case "Original series modified for extreme values"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".E1"
Case "Final adjustment difference"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".FAD"
Case "Record for outliers"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".ROT"
Case "Revision of seasonally adjusted data"
  datagraph$ = app.Path & "\output\" & seriesgraph & ".SFH"
Case "Revision of seasonal factors"
```

```
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".SAR"
   Case "Revision of period-to-period changes in seas.adj."
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".SAE"
   Case "Revision of trend-component"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".SFR"
   Case "Log-likelihood and AIC for observ."
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".SFE"
   Case "Mean forecast errors for observ."
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".CHR"
   Case "Revision of period-to-period changes in trend"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".CHE"
   Case "Revision of projected seasonal factors"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".TRR"
   Case "Record of seasonal filter selection"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".TRE"
   Case "Estimates of seasonally adjusted data"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".LKH"
   Case "Estimates of seasonal factors"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".FCE"
   Case "Estimates of period-to-period"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".TCR"
   Case "Estimates of trend component"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".TCE"
   Case "Estimates of period-to-period changes in trend"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".PSR"
   Case "Estimates of the projected seasonal factors"
    datagraph$ = app.Path & "\output\" & seriesgraph$ & ".PSE"

  End Select


End Sub

Sub list_graph ()
Dim a%

  a% = False
  If Val(getvar("formx12", "arguments0")) <> 0 Then
  If Val(getvar("series", "save0")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".A1") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments0")) <> 0 Then
  If Val(getvar("series", "save1")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".MVA") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx112")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".B1") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1117")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".E4") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1118")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".E5") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
```

```
 If Val(getvar("x11", "checkx1135")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B19") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1137")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C1") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1147")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".C19") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1149")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".D1") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1157")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E1") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments12")) <> 0 Then
 If Val(getvar("regadjust", "save5")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A11") <> "" Then
   a% = True
  End If
 End If
 End If
 If a% = False Then
  tables.Command3DGraph(0).Enabled = False
  Else
   tables.Command3DGraph(0).Enabled = True
 End If

 a% = False

 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save0")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A2") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save1")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A3") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save2")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".TRN") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments2")) <> 0 Then
 If Val(getvar("transform", "save3")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".A10") <> "" Then
```

```
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments12")) <> 0 Then
  If Val(getvar("regadjust", "save4")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".USR") <> "" Then
    a% = True
   End If
  End If
  End If

  If a = False Then
   tables.Command3DGraph(1).Enabled = False
   Else
    tables.Command3DGraph(1).Enabled = True
  End If

  a = False

  If Val(getvar("formx12", "arguments10")) <> 0 Then
  If Val(getvar("check", "save0")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".ACF") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments10")) <> 0 Then
  If Val(getvar("check", "save1")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".PCF") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments8")) <> 0 Then
  If Val(getvar("estimate", "checkestimate0")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".RSD") <> "" Then
    a% = True
   End If
  End If
  End If

  If a = False Then
   tables.Command3DGraph(2).Enabled = False
   Else
    tables.Command3DGraph(2).Enabled = True
  End If

  a = False

  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1110")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".D11") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1111")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".SAA") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1112")) <> 0 Then
   If Dir$(app.Path & "\output\" & seriesgraph & ".RND") <> "" Then
    a% = True
   End If
  End If
  End If
  If Val(getvar("formx12", "arguments3")) <> 0 Then
  If Val(getvar("x11", "checkx1116")) <> 0 Then
```

```
  If Dir$(app.Path & "\output\" & seriesgraph & ".FAD") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1117")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E4") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1119")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1120")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6A") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1121")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".E6R") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
 If Val(getvar("x11", "checkx1126")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph & ".B6") <> "" Then
   a% = True
  End If
 End If
 End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1130")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".B11") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1141")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C6") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1145")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C11") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1153")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".D6") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1158")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".E2") <> "" Then
   a% = True
       End If
```

```
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1160")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".E11") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1161")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".F1") <> "" Then
   a% = True
       End If
      End If
      End If

If a = False Then
 tables.Command3DGraph(3).Enabled = False
 Else
  tables.Command3DGraph(3).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1113")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".D12") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1122")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".E7") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1123")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".B2") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1127")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".B7") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1138")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C2") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1142")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C7") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1150")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".D2") <> "" Then
   a% = True
       End If
      End If
```

```
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1154")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".D7") <> "" Then
   a% = True
       End If
      End If
      End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1155")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".TAL") <> "" Then
   a% = True
       End If
      End If
      End If
If a = False Then
 tables.Command3DGraph(4).Enabled = False
 Else
  tables.Command3DGraph(4).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx114")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C17") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1114")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".D13") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1131")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".B13") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1133")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".B17") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1146")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".C13") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkxI156")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".IAO") <> "" Then
   a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1159")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph & ".E3") <> "" Then
   a% = True
       End If
      End If
      End If
If a = False Then
```

```
  tables.Command3DGraph(5).Enabled = False
  Else
    tables.Command3DGraph(5).Enabled = True
 End If

 a = False

 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx118")) <> 0 = True Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D10") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx119")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".FSD") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx1115")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D16") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx1125")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".B5") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx1129")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".B10") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx1140")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".C5") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx1152")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D5") <> "" Then
    a% = True
        End If
       End If
       End If
 If a = False Then
  tables.Command3DGraph(6).Enabled = False
  Else
    tables.Command3DGraph(6).Enabled = True
 End If

 a = False

 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx111")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".A4") <> "" Then
    a% = True
        End If
       End If
       End If
 If Val(getvar("formx12", "arguments3")) <> 0 Then
       If Val(getvar("x11", "checkx113")) <> 0 Then
```

```
      If Dir$(app.Path & "\output\" & seriesgraph & ".C16") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx115")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".C18") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx1115")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".D16") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx1132")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".B16") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx1134")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".B18") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx1135")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".B19") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("regadjust", "save2")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".TD") <> "" Then
   a% = True
      End If
     End If
     End If

If a = False Then
 tables.Command3DGraph(7).Enabled = False
 Else
  tables.Command3DGraph(7).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("x11", "checkx110")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".H1") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
     If Val(getvar("regadjust", "save3")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".HOL") <> "" Then
   a% = True
      End If
     End If
     End If

If a = False Then
 tables.Command3DGraph(8).Enabled = False
```

```
 Else
   tables.Command3DGraph(8).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx116")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D8") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx117")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D9") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1124")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".B3") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1128")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".B8") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1139")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".C4") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1143")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".C9") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments3")) <> 0 Then
      If Val(getvar("x11", "checkx1151")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".D4") <> "" Then
   a% = True
        End If
      End If
      End If

If a = False Then
  tables.Command3DGraph(9).Enabled = False
  Else
   tables.Command3DGraph(9).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments12")) <> 0 Then
      If Val(getvar("regadjust", "save0")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph & ".AO") <> "" Then
   a% = True
        End If
      End If
      End If
If Val(getvar("formx12", "arguments12")) <> 0 Then
      If Val(getvar("regadjust", "save1")) <> 0 Then
```

```
      If Dir$(app.Path & "\output\" & seriesgraph & ".LS") <> "" Then
   a% = True
      End If
     End If
     End If

If a = False Then
 tables.Command3DGraph(10).Enabled = False
 Else
  tables.Command3DGraph(10).Enabled = True
End If

a = False

If Val(getvar("formx12", "arguments11")) <> 0 Then
     If Val(getvar("forecast", "save0")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".FTR") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments11")) <> 0 Then
     If Val(getvar("forecast", "save1")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".FVR") <> "" Then
   a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments11")) <> 0 Then
     If Val(getvar("forecast", "save2")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph & ".FCT") <> "" Then
   a% = True
      End If
     End If
     End If

If a = False Then
 tables.Command3DGraph(11).Enabled = False
 Else
  tables.Command3DGraph(11).Enabled = True
End If

a% = False

If Val(getvar("formx12", "arguments13")) <> 0 Then
     If Val(getvar("sliding", "save0")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFS") <> "" Then
       a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
     If Val(getvar("sliding", "save1")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHS") <> "" Then
       a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
     If Val(getvar("sliding", "save2")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAS") <> "" Then
       a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
     If Val(getvar("sliding", "save3")) <> 0 Then
      If Dir$(app.Path & "\output\" & seriesgraph$ & ".YCS") <> "" Then
       a% = True
      End If
     End If
     End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
```

```
      If Val(getvar("sliding", "save4")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".TDS") <> "" Then
         a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
      If Val(getvar("sliding", "save5")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".SIS") <> "" Then
         a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
      If Val(getvar("sliding", "save6")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".CIS") <> "" Then
         a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
      If Val(getvar("sliding", "save7")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".AIS") <> "" Then
         a% = True
       End If
      End If
      End If
If Val(getvar("formx12", "arguments13")) <> 0 Then
      If Val(getvar("sliding", "save8")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".YIS") <> "" Then
         a% = True
       End If
      End If
      End If

  If a% = False Then
   tables.Command3DGraph(12).Enabled = False
   Else
    tables.Command3DGraph(12).Enabled = True
  End If

a% = False

      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save0")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".ROT") <> "" Then
         a% = True
       End If
      End If
      End If
      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save1")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFH") <> "" Then
         a% = True
       End If
      End If
      End If
      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save2")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAR") <> "" Then
         a% = True
       End If
      End If
      End If
      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save3")) <> 0 Then
       If Dir$(app.Path & "\output\" & seriesgraph$ & ".SAE") <> "" Then
         a% = True
       End If
      End If
      End If
      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save4")) <> 0 Then
```

```
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFR") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save5")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".SFE") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save6")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHR") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save7")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".CHE") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save8")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TRR") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save9")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TRE") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save10")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".LKH") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save11")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".FCE") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save12")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TCR") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save13")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".TCE") <> "" Then
    a% = True
  End If
 End If
End If
If Val(getvar("formx12", "arguments14")) <> 0 Then
If Val(getvar("history", "save14")) <> 0 Then
  If Dir$(app.Path & "\output\" & seriesgraph$ & ".PSR") <> "" Then
    a% = True
  End If
```

```
      End If
      End If
      If Val(getvar("formx12", "arguments14")) <> 0 Then
      If Val(getvar("history", "save15")) <> 0 Then
        If Dir$(app.Path & "\output\" & seriesgraph$ & ".PSE") <> "" Then
          a% = True
        End If
      End If
      End If

    If a% = False Then
      tables.Command3DGraph(13).Enabled = False
     Else
      tables.Command3DGraph(13).Enabled = True
    End If

End Sub
```

1.3. FICHEROS .TXT

1.3.1. Vtss.txt

```
Option Explicit
' vtss.txt
'
' Copyright (c) 1992-1994, VisualTools Inc.
'
' Contains constants and function declarations for
' accessing the Formula One 2.0 engine, VTSSDLL.DLL
'

Type POLYPOINT
   x As Integer
   y As Integer
End Type

Global Const kFormatRCNrBufSize = 16

Global Const kModeNormal = 0
Global Const kModeLine = 1
Global Const kModeRectangle = 2
Global Const kModeOval = 3
Global Const kModeArc = 4
Global Const kModeChart = 5
Global Const kModeField = 6
Global Const kModeButton = 7
Global Const kModePolygon = 8
Global Const kModeCheckBox = 9
Global Const kModeDropDown = 10

Global Const kPolyEditModeNormal = 0
Global Const kPolyEditModePoints = 1

Global Const kShiftHorizontal = 1
Global Const kShiftVertical = 2
Global Const kShiftRows = 3
Global Const kShiftCols = 4

Global Const kHAlignGeneral = 1
Global Const kHAlignLeft = 2
Global Const kHAlignCenter = 3
Global Const kHAlignRight = 4
Global Const kHAlignFill = 5
Global Const kHAlignJustify = 6
Global Const kHAlignCenterAcrossCells = 7

Global Const kVAlignTop = 1
Global Const kVAlignCenter = 2
Global Const kVAlignBottom = 3

Global Const kClearDlg = 0
Global Const kClearAll = 1
Global Const kClearFormats = 2
Global Const kClearValues = 3

Global Const kFileFormulaOne = 1
Global Const kFileExcel4 = 2
Global Const kFileTabbedText = 3      ' Only SSRead() supports this type.

Global Const kSSObjLine = 1
Global Const kSSObjRectangle = 2
Global Const kSSObjOval = 3
Global Const kSSObjArc = 4
Global Const kSSObjButton = 7
Global Const kSSObjPolygon = 8
Global Const kSSObjCheckBox = 9
Global Const kSSObjDropDown = 10
Global Const SSM_CHANGE = 1024
Global Const SSM_SELCHANGE = 1025
Global Const SSM_STARTEDIT = 1026
```

```
Global Const SSM_ENDEDIT = 1027
Global Const SSM_STARTRECALC = 1028
Global Const SSM_ENDRECALC = 1029
Global Const SSM_CLICK = 1030
Global Const SSM_DBLCLICK = 1031
Global Const SSM_DELETEROW = 1032
Global Const SSM_REFRESHROW = 1033
Global Const SSM_CELLCHANGED = 1034
Global Const SSM_SETFOCUS = 1035
Global Const SSM_MODIFIED = 1036
Global Const SSM_DELETETABLE = 1037
Global Const SSM_HELP = 1038
Global Const SSM_DATAMOVE = 1039
Global Const SSM_CANCELEDIT = 1040
Global Const SSM_DIALOGBOXPARAM = 1041
Global Const SSM_FETCH = 1042
Global Const SSM_GETDATAROWS = 1043
Global Const SSM_TOPLEFTCHANGED = 1044
Global Const SSM_OBJCLICK = 1045
Global Const SSM_OBJDBLCLICK = 1046
Global Const SSM_RCLICK = 1047
Global Const SSM_RDBLCLICK = 1048

Global Const SSERROR_NONE = 0
Global Const SSERROR_GENERAL = 1
Global Const SSERROR_BAD_ARGUMENT = 2
Global Const SSERROR_NO_MEMORY = 3
Global Const SSERROR_BAD_FORMULA = 4
Global Const SSERROR_BUF_TOO_SHORT = 5
Global Const SSERROR_NOT_FOUND = 6
Global Const SSERROR_BAD_RC = 7
Global Const SSERROR_BAD_HSS = 8
Global Const SSERROR_TOO_MANY_HSS = 9
Global Const SSERROR_NO_TABLE = 10
Global Const SSERROR_UNABLE_TO_OPEN_FILE = 11
Global Const SSERROR_INVALID_FILE = 12
Global Const SSERROR_INSERT_SHIFT_OFF_TABLE = 13
Global Const SSERROR_ONLY_ONE_RANGE = 14
Global Const SSERROR_NOTHING_TO_PASTE = 15
Global Const SSERROR_BAD_NUMBER_FORMAT = 16
Global Const SSERROR_TOO_MANY_FONTS = 17
Global Const SSERROR_TOO_MANY_SELECTED_RANGES = 18
Global Const SSERROR_UNABLE_TO_WRITE_FILE = 19
Global Const SSERROR_NO_TRANSACTION = 20
Global Const SSERROR_NOTHING_TO_PRINT = 21
Global Const SSERROR_PRINT_MARGINS_DONT_FIT = 22
Global Const SSERROR_CANCEL = 23
Global Const SSERROR_UNABLE_TO_INITIALIZE_PRINTER = 24
Global Const SSERROR_STRING_TOO_LONG = 25
Global Const SSERROR_FORMULA_TOO_LONG = 26
Global Const SSERROR_UNABLE_TO_OPEN_CLIPBOARD = 27
Global Const SSERROR_PASTE_WOULD_OVERFLOW_SHEET = 28
Global Const SSERROR_LOCKED_CELLS_CANNOT_BE_MODIFIED = 29
Global Const SSERROR_LOCKED_DOC_CANNOT_BE_MODIFIED = 30
Global Const SSERROR_INVALID_NAME = 31
Global Const SSERROR_CANT_DELETE_NAME_IN_USE = 32
Global Const SSERROR_UNABLE_TO_FIND_NAME = 33
Global Const SSERROR_NO_WINDOW = 34
Global Const SSERROR_SELECTION = 35
Global Const SSERROR_TOO_MANY_OBJECTS = 36
Global Const SSERROR_INVALID_OBJECT_TYPE = 37
Global Const SSERROR_OBJECT_NOT_FOUND = 38
Global Const SSERROR_INVALID_REQUEST = 39

Declare Function SSAddColPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%)
Declare Function SSAddPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSAddRowPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%)
Declare Function SSAddSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal
nC2%)
Declare Function SSAttach% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pTitle$)
Declare Function SSAttachToSS% Lib "VTSSDLL.DLL" (ByVal hDstSS&, ByVal hSrcSS&)
Declare Function SSCalculationDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
```

```
Declare Function SSCallWindowProc% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMessage%, ByVal wParam%, ByVal
lParam&)
Declare Function SSCancelEdit% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSCanEditPaste% Lib "VTSSDLL.DLL" (ByVal hSS&, pCanEditPaste%)
Declare Function SSEditPasteValues% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSCheckModified% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSCheckRecalc% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSClearClipboard% Lib "VTSSDLL.DLL" ()
Declare Function SSClearRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%,
ByVal nClearType%)
Declare Function SSColorPaletteDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSColWidthDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSCopyAll% Lib "VTSSDLL.DLL" (ByVal hDstSS&, ByVal hSrcSS&)
Declare Function SSCopyRange% Lib "VTSSDLL.DLL" (ByVal hDstSS&, ByVal nDstR1%, ByVal nDstC1%, ByVal nDstR2%,
ByVal nDstC2%, ByVal hSrcSS&, ByVal nSrcR1%, ByVal nSrcC1%, ByVal nSrcR2%, ByVal nSrcC2%)
Declare Function SSDefinedNameDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSDelete% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bSendDeleteTableMsg%)
Declare Function SSDeleteDefinedName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pName$)
Declare Function SSDeleteRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%,
ByVal nShiftType%)
Declare Function SSDeleteTable% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEditBarDelete% Lib "VTSSDLL.DLL" (ByVal hSSEdit&)
Declare Function SSEditBarHeight% Lib "VTSSDLL.DLL" ()
Declare Function SSEditBarMove% Lib "VTSSDLL.DLL" (ByVal hSSEdit&, ByVal x%, ByVal y%, ByVal cx%, ByVal cy%)
Declare Function SSEditBarNew% Lib "VTSSDLL.DLL" (ByVal hWndParent%, pSSEdit&)
Declare Function SSEditClear% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nClearType%)
Declare Function SSEditCopy% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEditCopyDown% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEditCopyRight% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEditCut% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEditDelete% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nShiftType%)
Declare Function SSEditInsert% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nShiftType%)
Declare Function SSEditPaste% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSEndEdit% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSErrorNumberToText% Lib "VTSSDLL.DLL" (ByVal nError%, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSFilePageSetupDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFilePrint% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bShowPrintDlg%)
Declare Function SSFilePrintSetupDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatAlignmentDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatBorderDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatCurrency0% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatCurrency2% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatFixed% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatFixed2% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatFontDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatFraction% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatGeneral% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatHmmampm% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatMdyy% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatNumberDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatPatternDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatPercent% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSFormatRCNr% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal bDoAbsolute%,
ByVal pBuf$, ByVal nBufSize%)
Declare Function SSFormatScientific% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSGetActiveCell% Lib "VTSSDLL.DLL" (ByVal hSS&, pRow%, pCol%)
Declare Function SSGetAlignment% Lib "VTSSDLL.DLL" (ByVal hSS&, pHorizontal%, pWordWrap%, pVertical%,
pOrientation%)
Declare Function SSGetAllowArrows% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowArrows%)
Declare Function SSGetAllowDelete% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowDelete%)
Declare Function SSGetAllowEditHeaders% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowEditHeaders%)
Declare Function SSGetAllowFillRange% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowFillRange%)
Declare Function SSGetAllowInCellEditing% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowInCellEditing%)
Declare Function SSGetAllowMoveRange% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowMoveRange%)
Declare Function SSGetAllowObjSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowObjSelections%)
Declare Function SSGetAllowResize% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowResize%)
Declare Function SSGetAllowSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowSelections%)
Declare Function SSGetAllowTabs% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowTabs%)
Declare Function SSGetAllowFormulas% Lib "VTSSDLL.DLL" (ByVal hSS&, pAllowFormulas%)
Declare Function SSGetAppName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetAutoRecalc% Lib "VTSSDLL.DLL" (ByVal hSS&, pAutoRecalc%)
Declare Function SSGetBackColor% Lib "VTSSDLL.DLL" (ByVal hSS&, pBackColor&)
```

```
Declare Function SSGetBorder% Lib "VTSSDLL.DLL" (ByVal hSS&, pLeft%, pRight%, pTop%, pBottom%, pShade%, pcrLeft&,
pcrRight&, pcrTop&, pcrBottom&)
Declare Function SSGetColText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetColWidth% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%, pWidth%)
Declare Function SSGetDefinedName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pName$, ByVal Buf$, ByVal nBufSize%)
Declare Function SSGetDoSetCursor% Lib "VTSSDLL.DLL" (ByVal hSS&, pDoSetCursor%)
Declare Function SSGetEnableProtection% Lib "VTSSDLL.DLL" (ByVal hSS&, pEnableProtection%)
Declare Function SSGetEnterMovesDown% Lib "VTSSDLL.DLL" (ByVal hSS&, pEnterMovesDown%)
Declare Function SSGetEntry% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetEntryRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal Buf$, ByVal
nBufSize%)
Declare Function SSGetExtraColor% Lib "VTSSDLL.DLL" (ByVal hSS&, pExtraColor&)
Declare Function SSGetFireEvent% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nEvent%, pFireIt%)
Declare Function SSGetFixedCols% Lib "VTSSDLL.DLL" (ByVal hSS&, pC1%, pCols%)
Declare Function SSGetFixedRows% Lib "VTSSDLL.DLL" (ByVal hSS&, pR1%, pRows%)
Declare Function SSGetFont% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%, pSize%, pBold%, pItalic%,
pUnderline%, pStrikeout%, pcrColor&, pOutline%, pShadow%)
Declare Function SSGetFormattedText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetFormattedTextRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal pBuf$,
ByVal nBufSize%)
Declare Function SSGetFormula% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetFormulaRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal Buf$, ByVal
nBufSize%)
Declare Function SSGetHdrHeight% Lib "VTSSDLL.DLL" (ByVal hSS&, pHeight%)
Declare Function SSGetHdrSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, pTopLeftHdr%, pRowHdr%, pColHdr%)
Declare Function SSGetHdrWidth% Lib "VTSSDLL.DLL" (ByVal hSS&, pWidth%)
Declare Function SSGetIteration% Lib "VTSSDLL.DLL" (ByVal hSS&, pIteration%, pMaxIterations%, pMaxChange#)
Declare Function SSGetLastCol% Lib "VTSSDLL.DLL" (ByVal hSS&, pLastCol%)
Declare Function SSGetLastColForRow% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, pLastColForRow%)
Declare Function SSGetLastRow% Lib "VTSSDLL.DLL" (ByVal hSS&, pLastRow%)
Declare Function SSGetLeftCol% Lib "VTSSDLL.DLL" (ByVal hSS&, pLeftCol%)
Declare Function SSGetLineStyle% Lib "VTSSDLL.DLL" (ByVal hSS&, pStyle%, pcrColor&, pWeight%)
Declare Function SSGetLogicalRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, pIsTrue%)
Declare Function SSGetMinCol% Lib "VTSSDLL.DLL" (ByVal hSS&, pMinCol%)
Declare Function SSGetMinRow% Lib "VTSSDLL.DLL" (ByVal hSS&, pMinRow%)
Declare Function SSGetMaxCol% Lib "VTSSDLL.DLL" (ByVal hSS&, pMaxCol%)
Declare Function SSGetMaxRow% Lib "VTSSDLL.DLL" (ByVal hSS&, pMaxRow%)
Declare Function SSGetMode% Lib "VTSSDLL.DLL" (ByVal hSS&, pMode%)
Declare Function SSGetNumber% Lib "VTSSDLL.DLL" (ByVal hSS&, pNumber#)
Declare Function SSGetNumberFormat% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetNumberRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, pNumber#)
Declare Function SSGetPattern% Lib "VTSSDLL.DLL" (ByVal hSS&, pPattern%, pcrFG&, pcrBG&)
Declare Function SSGetPolyEditMode% Lib "VTSSDLL.DLL" (ByVal hSS&, pPolyEditMode%)
Declare Function SSGetPrintArea% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetPrintBottomMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintBottomMargin#)
Declare Function SSGetPrintColHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintColHeading%)
Declare Function SSGetPrintFooter% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetPrintGridLines% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintGridLines%)
Declare Function SSGetPrintHCenter% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintHCenter%)
Declare Function SSGetPrintHeader% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetPrintLeftMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintLeftMargin#)
Declare Function SSGetPrintLeftToRight% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintLeftToRight%)
Declare Function SSGetPrintNoColor% Lib "VTSSDLL.DLL" (ByVal hSS&, pNoColor%)
Declare Function SSGetPrintRightMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintRightMargin#)
Declare Function SSGetPrintRowHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintRowHeading%)
Declare Function SSGetPrintScale% Lib "VTSSDLL.DLL" (ByVal hSS&, pScale%, pFitToPage%, pVPages%, pHPages%)
Declare Function SSGetPrintTitles% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetPrintTopMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintTopMargin#)
Declare Function SSGetPrintVCenter% Lib "VTSSDLL.DLL" (ByVal hSS&, pPrintVCenter%)
Declare Function SSGetProtection% Lib "VTSSDLL.DLL" (ByVal hSS&, pLocked%, pHidden%)
Declare Function SSGetRepaint% Lib "VTSSDLL.DLL" (ByVal hSS&, pRepaint%)
Declare Function SSGetRowHeight% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, pHeight%)
Declare Function SSGetRowMode% Lib "VTSSDLL.DLL" (ByVal hSS&, pRowMode%)
Declare Function SSGetRowText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetScale% Lib "VTSSDLL.DLL" (ByVal hSS&, pScale%)
Declare Function SSGetSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nSelection%, pR1%, pC1%, pR2%, pC2%)
Declare Function SSGetSelectionCount% Lib "VTSSDLL.DLL" (ByVal hSS&, pCount%)
Declare Function SSGetSelectionRef% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetShowColHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowColHeading%)
Declare Function SSGetShowFormulas% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowFormulas%)
Declare Function SSGetShowGridLines% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowGridLines%)
Declare Function SSGetShowHScrollBar% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowHScrollBar%)
Declare Function SSGetShowRowHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowRowHeading%)
```

```
Declare Function SSGetShowSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowSelections%)
Declare Function SSGetShowVScrollBar% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowVScrollBar%)
Declare Function SSGetShowZeroValues% Lib "VTSSDLL.DLL" (ByVal hSS&, pShowZeroValues%)
Declare Function SSGetSSEdit% Lib "VTSSDLL.DLL" (ByVal hSS&, pSSEdit&)
Declare Function SSGetTabbedText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal
nC2%, ByVal bValuesOnly%, phText%)
Declare Function SSGetText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetTextRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal pBuf$, ByVal
nBufSize%)
Declare Function SSGetTitle% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetTopLeftText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSGetTopRow% Lib "VTSSDLL.DLL" (ByVal hSS&, pTopRow%)
Declare Function SSGetTypeRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, pType%)
Declare Function SSGotoDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSInitTable% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSInsertRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%,
ByVal nShiftType%)
Declare Function SSLaunchAppDesigner% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSLineStyleDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSMaxCol% Lib "VTSSDLL.DLL" ()
Declare Function SSMaxRow% Lib "VTSSDLL.DLL" ()
Declare Function SSMoveRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%,
ByVal nRowOffset%, ByVal nColOffset%)
Declare Function SSNew% Lib "VTSSDLL.DLL" (ByVal hWnd%, pSS&)
Declare Function SSNextColPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%, pNextCol%)
Declare Function SSNextRowPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, pNextRow%)
Declare Function SSObjAddItem% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pItem$)
Declare Function SSObjAddSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&)
Declare Function SSObjBringToFront% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSObjDeleteItem% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nItem%)
Declare Function SSObjFirstID% Lib "VTSSDLL.DLL" (ByVal hSS&, pFirstID&)
Declare Function SSObjGetCell% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pHasCell%, pRow%, pCol%)
Declare Function SSObjGetItem% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nItem%, ByVal pBuf$, ByVal
nBufSize%)
Declare Function SSObjGetItemCount% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pItems%)
Declare Function SSObjGetItems% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSObjGetName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSObjGetPos% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pX1!, pY1!, pX2!, pY2!)
Declare Function SSObjGetSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nSelection%, pID&)
Declare Function SSObjGetSelectionCount% Lib "VTSSDLL.DLL" (ByVal hSS&, pCount%)
Declare Function SSObjGetText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pBuf$, ByVal nBufSize%)
Declare Function SSObjGetType% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pType%)
Declare Function SSObjGetValue% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pValue%)
Declare Function SSObjGetVisible% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pVisible%)
Declare Function SSObjInsertItem% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nItem%, ByVal pItem$)
Declare Function SSObjNameDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSObjNameToID% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pName$, pID&)
Declare Function SSObjNew% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nType%, ByVal nX1!, ByVal nY1!, ByVal nX2!,
ByVal nY2!, pID&)
Declare Function SSObjNewPicture% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nX1!, ByVal nY1!, ByVal nX2!, ByVal nY2!,
pID&, ByVal hMF%, ByVal nMapMode%, ByVal nWndExtentX%, ByVal nWndExtentY%)
Declare Function SSObjNewPolygon% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nX1!, ByVal nY1!, ByVal nX2!, ByVal nY2!,
pID&, pPoints As POLYPOINT, ByVal nPoints%, ByVal bClosed%)
Declare Function SSObjNextID% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pNextID&)
Declare Function SSObjOptionsDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSObjPosToTwips% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nX1!, ByVal nY1!, ByVal nX2!, ByVal nY2!,
pX&, pY&, pCX&, pCY&, pShown%)
Declare Function SSObjSendToBack% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSObjSetCell% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal bHasCell%, ByVal nRow%, ByVal
nCol%)
Declare Function SSObjSetItem% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nItem%, ByVal pItem$)
Declare Function SSObjSetItems% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pItems$)
Declare Function SSObjSetName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pName$)
Declare Function SSObjSetPicture% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal hMF%, ByVal nMapMode%, ByVal
nWndExtentX%, ByVal nWndExtentY%)
Declare Function SSObjSetPolygonPoints% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, pPoints As POLYPOINT, ByVal
nPoints%, ByVal bClosed%)
Declare Function SSObjSetPos% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nX1!, ByVal nY1!, ByVal nX2!, ByVal
nY2!)
Declare Function SSObjSetSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&)
Declare Function SSObjSetText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal pText$)
Declare Function SSObjSetValue% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal nValue%)
Declare Function SSObjSetVisible% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal ID&, ByVal bVisible%)
```

```
Declare Function SSOpenFileDlg% Lib "VTSSDLL.DLL" (ByVal pTitle$, ByVal hWndParent%, ByVal pBuf$, ByVal
nBufSize%)
Declare Function SSProtectionDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSRangeToTwips% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow1%, ByVal nCol1%, ByVal nRow2%,
ByVal nCol2%, pX&, pY&, pCX&, pCY&, pShown%)
Declare Function SSRead% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pPathName$, pFileType%)
Declare Function SSReadIO% Lib "VTSSDLL.DLL" (ByVal hSS&, dwUserData&, ByVal ioFunc&, pUserRet&)
Declare Function SSRecalc% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSRemoveColPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%)
Declare Function SSRemovePageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSRemoveRowPageBreak% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%)
Declare Function SSRowHeightDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSSaveFileDlg% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pTitle$, ByVal pBuf$, ByVal nBufSize%,
pFileType%)
Declare Function SSSaveWindowInfo% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSSetActiveCell% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%)
Declare Function SSSetAlignment% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nHorizontal%, ByVal bWordWrap%, ByVal
nVertical%, ByVal nOrientation%)
Declare Function SSSetAllowArrows% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowArrows%)
Declare Function SSSetAllowDelete% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowDelete%)
Declare Function SSSetAllowEditHeaders% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowEditHeaders%)
Declare Function SSSetAllowFillRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowFillRange%)
Declare Function SSSetAllowInCellEditing% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowInCellEditing%)
Declare Function SSSetAllowMoveRange% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowMoveRange%)
Declare Function SSSetAllowObjSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowObjSelections%)
Declare Function SSSetAllowResize% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowResize%)
Declare Function SSSetAllowSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowSelections%)
Declare Function SSSetAllowTabs% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowTabs%)
Declare Function SSSetAllowFormulas% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAllowFormulas%)
Declare Function SSSetAutoRecalc% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bAutoRecalc%)
Declare Function SSSetAppName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pAppName$)
Declare Function SSSetBackColor% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal crBackColor&)
Declare Function SSSetBorder% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nOutline%, ByVal nLeft%, ByVal nRight%, ByVal
nTop%, ByVal nBottom%, ByVal nShade%, ByVal crOutline&, ByVal crLeft&, ByVal crRight&, ByVal crTop&, ByVal
crBottom&)
Declare Function SSSetColText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nCol%, ByVal pColText$)
Declare Function SSSetColWidth% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nC1%, ByVal nC2%, ByVal nWidth%, ByVal
bDefColWidth%)
Declare Function SSSetColWidthAuto% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal
nC2%, ByVal bSetDefaults%)
Declare Function SSSetDefinedName% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pName$, ByVal pFormula$)
Declare Function SSSetDefWindowProc% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pWindowProc&)
Declare Function SSSetDoSetCursor% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bDoSetCursor%)
Declare Function SSSetEnableProtection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bEnableProtection%)
Declare Function SSSetEnterMovesDown% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bEnterMovesDown%)
Declare Function SSSetEntry% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pEntry$)
Declare Function SSSetEntryRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal pEntry$)
Declare Function SSSetExtraColor% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal crExtraColor&)
Declare Function SSSetFireEvent% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nEvent%, ByVal bFireIt%)
Declare Function SSSetFixedCols% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nC1%, ByVal nCols%)
Declare Function SSSetFixedRows% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nRows%)
Declare Function SSSetFont% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pName$, ByVal nSize%, ByVal bBold%, ByVal
bItalic%, ByVal bUnderline%, ByVal bStrikeout%, ByVal crColor&, ByVal bOutline%, ByVal bShadow%)
Declare Function SSSetFormula% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pFormula$)
Declare Function SSSetFormulaRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal pFormula$)
Declare Function SSSetHdrHeight% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nHeight%)
Declare Function SSSetHdrSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bTopLeftHdr%, ByVal bRowHdr%, ByVal
bColHdr%)
Declare Function SSSetHdrWidth% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nWidth%)
Declare Function SSSetIteration% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bIteration%, ByVal nMaxIterations%, ByVal
nMaxChange#)
Declare Function SSSetLeftCol% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nLeftCol%)
Declare Function SSSetLineStyle% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nStyle%, ByVal crColor&, ByVal nWeight%)
Declare Function SSSetLogicalRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal bIsTrue%)
Declare Function SSSetMinCol% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMinCol%)
Declare Function SSSetMinRow% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMinRow%)
Declare Function SSSetMaxCol% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMaxCol%)
Declare Function SSSetMaxRow% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMaxRow%)
Declare Function SSSetMode% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMode%)
Declare Function SSSetNumber% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nNumber#)
Declare Function SSSetNumberFormat% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pNumberFormat$)
Declare Function SSSetNumberRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal nNumber#)
Declare Function SSSetPattern% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nPattern%, ByVal crFG&, ByVal crBG&)
```

```
Declare Function SSSetPolyEditMode% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nPolyEditMode%)
Declare Function SSSetPrintArea% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pFormula$)
Declare Function SSSetPrintAreaFromSelection% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSSetPrintBottomMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMargin#)
Declare Function SSSetPrintColHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bColHeading%)
Declare Function SSSetPrintFooter% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pPrintFooter$)
Declare Function SSSetPrintGridLines% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bGridLines%)
Declare Function SSSetPrintHCenter% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bHCenter%)
Declare Function SSSetPrintHeader% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pPrintHeader$)
Declare Function SSSetPrintLeftMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMargin#)
Declare Function SSSetPrintLeftToRight% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bLeftToRight%)
Declare Function SSSetPrintNoColor% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bNoColor%)
Declare Function SSSetPrintRightMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMargin#)
Declare Function SSSetPrintRowHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bRowHeading%)
Declare Function SSSetPrintScale% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nScale%, ByVal bFitToPage%, ByVal
nVPages%, ByVal nHPages%)
Declare Function SSSetPrintTitles% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pFormula$)
Declare Function SSSetPrintTitlesFromSelection% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSSetPrintTopMargin% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nMargin#)
Declare Function SSSetPrintVCenter% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bVCenter%)
Declare Function SSSetProtection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bLocked%, ByVal bHidden%)
Declare Function SSSetRepaint% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bRepaint%)
Declare Function SSSetRowHeight% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nR2%, ByVal nHeight%, ByVal
bDefRowHeight%)
Declare Function SSSetRowHeightAuto% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal
nC2%, ByVal bSetDefaults%)
Declare Function SSSetRowMode% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bRowMode%)
Declare Function SSSetRowText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal pRowText$)
Declare Function SSSetScale% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nScale%)
Declare Function SSSetSelection% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%)
Declare Function SSSetSelectionRef% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pFormula$)
Declare Function SSSetShowColHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bColHeading%)
Declare Function SSSetShowFormulas% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bFormulas%)
Declare Function SSSetShowGridLines% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bGridLines%)
Declare Function SSSetShowHScrollBar% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nShowHScrollBar%)
Declare Function SSSetShowRowHeading% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bRowHeading%)
Declare Function SSSetShowSelections% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nSelections%)
Declare Function SSSetShowVScrollBar% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nShowVScrollBar%)
Declare Function SSSetShowZeroValues% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bZeroValues%)
Declare Function SSSetSSEdit% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal hSSEdit&)
Declare Function SSSetTabbedText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nStartRow%, ByVal nStartCol%, pRows%,
pCols%, ByVal bValuesOnly%, ByVal pText$)
Declare Function SSSetText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pText$)
Declare Function SSSetTextRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nRow%, ByVal nCol%, ByVal pText$)
Declare Function SSSetTitle% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pTitle$)
Declare Function SSSetTopLeftText% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pTopLeftText$)
Declare Function SSSetTopRow% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nTopRow%)
Declare Function SSShowActiveCell% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSSort% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%, ByVal
bSortByRows%, pKeys%, ByVal nKeys%)
Declare Function SSSort3% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%, ByVal nC2%, ByVal
bSortByRows%, ByVal nKey1%, ByVal nKey2%, ByVal nKey3%)
Declare Function SSSortDlg% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSStartEdit% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal bClear%, ByVal bInCellEditFocus%, ByVal
bArrowsExitEditMode%)
Declare Function SSSwapTables% Lib "VTSSDLL.DLL" (ByVal hSS1&, ByVal hSS2&)
Declare Function SSTransactCommit% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSTransactRollBack% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSTransactStart% Lib "VTSSDLL.DLL" (ByVal hSS&)
Declare Function SSTwipsToRC% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal x&, ByVal y&, pRow%, pCol%)
Declare Sub SSUpdate Lib "VTSSDLL.DLL" ()
Declare Function SSVersion% Lib "VTSSDLL.DLL" ()
Declare Function SSWrite% Lib "VTSSDLL.DLL" (ByVal hSS&, ByVal pPathName$, ByVal nFileType%)
Declare Function SSWriteIO% Lib "VTSSDLL.DLL" (ByVal hSS&, dwUserData&, ByVal ioFunc&, pUserRet&)

Declare Function SSVBXCopyCellsFromDoubleArray% Lib "VTSS.VBX" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal
nR2%, ByVal nC2%, nArray() As Double)
Declare Function SSVBXCopyCellsToDoubleArray% Lib "VTSS.VBX" (ByVal hSS&, ByVal nR1%, ByVal nC1%, ByVal nR2%,
ByVal nC2%, nArray() As Double)
```

1.3.2. Vtchvb.txt

```
'
' vtchvb.txt
'
' Copyright (c) 1994, VisualTools Inc.
'


'
' from vtdefs.h
'


' VtHorizontalAlignment
'     Defines the types of horizontal text alignment
'
Global Const VtHorizontalAlignmentLeft = 0
Global Const VtHorizontalAlignmentRight = 1
Global Const VtHorizontalAlignmentCenter = 2


' VtVerticalAlignment
'     Defines the types of vertical text alignment
'
Global Const VtVerticalAlignmentTop = 0
Global Const VtVerticalAlignmentBottom = 1
Global Const VtVerticalAlignmentCenter = 2


' VtOrientation
'     Defines the orientation of the text
'
Global Const VtOrientationHorizontal = 0
Global Const VtOrientationVertical = 1
Global Const VtOrientationUp = 2
Global Const VtOrientationDown = 3


' VtSortType
'
Global Const VtSortTypeNone = 0
Global Const VtSortTypeAscending = 1
Global Const VtSortTypeDescending = 2


' VtAngleUnits
'     Describes the type of angle units
'
Global Const VtAngleUnitsDegrees = 0
Global Const VtAngleUnitsRadians = 1
Global Const VtAngleUnitsGrads = 2


'
' from vterrors.h
'


' VtErrorNumber
'         error codes
```

```
'

Global Const VtOk = 0

' internal errors
Global Const VtFail = 1000
Global Const VtErrorDeletingUsedObject = 1001
Global Const VtErrorCorruptData = 1002
Global Const VtErrorNotImplemented = 1003

' general errors
Global Const VtErrorNoMemory = 1100
Global Const VtErrorInvalidArgument = 1101
Global Const VtErrorNotFound = 1102
Global Const VtErrorTooSmall = 1103
Global Const VtErrorInvalidRequest = 1104
Global Const VtErrorCorruptArchive = 1105              ' persistence storage corrupted
Global Const VtErrorArchiveVersion = 1106              ' tried to load incompatible version
Global Const VtErrorArchiveTypeMismatch = 1107             ' unexpected class type
Global Const VtErrorArchivePointerMismatch = 1108          ' tried to load ptr into reference
Global Const VtErrorCannotOpenFile = 1109
Global Const VtErrorUnableToLoadString = 1110
Global Const VtErrorBufferTooSmall = 1111

' display driver errors
Global Const VtErrorDisplay = 1500
Global Const VtErrorInvalidFontName = 1501
Global Const VtErrorInvalidFont = 1502
Global Const VtErrorNoDisplayResources = 1503

' chart errors
Global Const VtChError = 2000
Global Const VtChErrorInvalidHandle = 2001
Global Const VtChErrorNoData = 2002
Global Const VtChErrorInvalidSeriesNumber = 2003
Global Const VtChErrorInvalidAxis = 2004


'''''''
```

```
End Type

'-----------------------------------------------------------------------
' VtcCoor3
'
Type VtcCoor3
        x As Single
        y As Single
        z As Single
End Type

'-----------------------------------------------------------------------
' VtcLCoor
'
Type VtcLCoor
        x As Long
        y As Long
End Type

'-----------------------------------------------------------------------
' VtcLRect
'
Type VtcLRect
        min As VtcLCoor
        max As VtcLCoor
End Type

'-----------------------------------------------------------------------
' VtcColor
Type VtcColor
        red As Integer
        green As Integer
        blue As Integer
        automatic As Integer
End Type

'-----------------------------------------------------------------------
' VtcFont
'       font description
'
'  Font Styles
Global Const VtFontStyleBold = &H1
Global Const VtFontStyleItalic = &H2
Global Const VtFontStyleOutline = &H4

'  Font Effects
Global Const VtFontEffectStrikeThrough = &H100
Global Const VtFontEffectUnderline = &H200

Type VtcFont
        name As String * 48
        size As Single              ' point size
        style As Integer            ' see VtFontStyle's above
        effects As Integer          ' see VtFontEffect's above
        color As VtcColor
End Type

'-----------------------------------------------------------------------
' VtcBrush
'
' Brush Style
Global Const VtBrushStyleNull = 0
Global Const VtBrushStyleSolid = 1
Global Const VtBrushStylePattern = 2
```

```
Global Const VtBrushStyleHatched = 3

' Brush patterns

Global Const VtBrushPattern94Percent = 0
Global Const VtBrushPattern88Percent = 1
Global Const VtBrushPattern75Percent = 2
Global Const VtBrushPattern50Percent = 3
Global Const VtBrushPattern25Percent = 4
Global Const VtBrushPatternBoldHorizontal = 5
Global Const VtBrushPatternBoldVertical = 6
Global Const VtBrushPatternBoldDownDiagonal = 7
Global Const VtBrushPatternBoldUpDiagonal = 8
Global Const VtBrushPatternChecks = 9
Global Const VtBrushPatternWeave = 10
Global Const VtBrushPatternHorizontal = 11
Global Const VtBrushPatternVertical = 12
Global Const VtBrushPatternDownDiagonal = 13
Global Const VtBrushPatternUpDiagonal = 14
Global Const VtBrushPatternGrid = 15
Global Const VtBrushPatternTrellis = 16
Global Const VtBrushPatternInvertedTrellis = 17

Global Const VtBrushPatternCount = 18

' Brush Hatches

Global Const VtBrushHatchHorizontal = 0
Global Const VtBrushHatchVertical = 1
Global Const VtBrushHatchDownDiagonal = 2
Global Const VtBrushHatchUpDiagonal = 3
Global Const VtBrushHatchCross = 4
Global Const VtBrushHatchDiagonalCross = 5

Global Const VtBrushHatchCount = 6


Type VtcBrush
        style As Integer
        fillColor As VtcColor
        patternColor As VtcColor
        index As Integer
End Type

'---------------------------------------------------------------------

' VtcShadow
'

Global Const VtShadowStyleNull = 0
Global Const VtShadowStyleDrop = 1

Type VtcShadow
        style As Integer
        brush As VtcBrush
        offset As VtcCoor
End Type



'---------------------------------------------------------------------

' VtcPen
'

'    pen styles
'

Global Const VtPenStyleNull = 0
Global Const VtPenStyleSolid = 1
Global Const VtPenStyleDashed = 2
Global Const VtPenStyleDotted = 3
Global Const VtPenStyleDashDot = 4
Global Const VtPenStyleDashDotDot = 5
Global Const VtPenStyleDitted = 6
Global Const VtPenStyleDashDit = 7
```

```
Global Const VtPenStyleDashDitDit = 8
Global Const VtPenStyleNative = 9

'     pen join types
'
Global Const VtPenJoinMiter = 0
Global Const VtPenJoinRound = 1
Global Const VtPenJoinBevel = 2

'     pen cap types
'
Global Const VtPenCapButt = 0
Global Const VtPenCapRound = 1
Global Const VtPenCapSquare = 2

Type VtcPen
        style As Integer
        join As Integer
        cap As Integer
        color As VtcColor
        width As Single
        limit As Single
End Type


'-----------------------------------------------------------------------
' VtcFrame
'

'     frame styles
Global Const VtFrameStyleNull = 0
Global Const VtFrameStyleSingleLine = 1
Global Const VtFrameStyleDoubleLine = 2
Global Const VtFrameStyleThickInner = 3
Global Const VtFrameStyleThickOuter = 4

Type VtcFrame
        style As Integer
        color As VtcColor
        spaceColor As VtcColor
        width As Single
End Type


'-----------------------------------------------------------------------
' VtcPicture
'

'     picture types
Global Const VtPictureTypeNull = 0
Global Const VtPictureTypeBitmap = 1    ' Implementation dependent bitmap
Global Const VtPictureTypeVector = 2    ' Implementation dependent vector
Global Const VtPictureTypeBMP = 3
Global Const VtPictureTypeWMF = 4           ' Windows Metafile
Global Const VtPictureTypeGIF = 6
Global Const VtPictureTypePCX = 7
Global Const VtPictureTypeTIFF = 8
Global Const VtPictureTypePICT = 9

' picture options
Global Const VtPictureOptionNoSizeHeader = 1
Global Const VtPictureOptionTextAsCurves = 2

'     picture mapping options
Global Const VtPictureMapTypeActual = 0
Global Const VtPictureMapTypeFitted = 1
Global Const VtPictureMapTypeStretched = 2
Global Const VtPictureMapTypeTiled = 3
Global Const VtPictureMapTypeCropFitted = 4

Type VtcPicture
        type As Integer
```

```
        map As Integer
        embedded As Integer
        name As String * 64
End Type

'---------------------------------------------------------------------------
' VtcGradient
'
Global Const VtGradientStyleHorizontal = 0
Global Const VtGradientStyleVertical = 1
Global Const VtGradientStyleRectangle = 2
Global Const VtGradientStyleOval = 3

Type VtcGradient
        style As Integer
        fromColor As VtcColor
        toColor As VtcColor
End Type

'---------------------------------------------------------------------------
' VtcFill
'
Global Const VtFillStyleNull = 0        ' no brush or gradient fill
Global Const VtFillStyleBrush = 1
Global Const VtFillStyleGradient = 2

Type VtcFill
        style As Integer
        brush As VtcBrush
        gradient As VtcGradient
        picture As VtcPicture
End Type


'---------------------------------------------------------------------------
' VtcBackdrop
'
Type VtcBackdrop
        shadow As VtcShadow
        frame As VtcFrame
        fill As VtcFill
End Type


'---------------------------------------------------------------------------
' VtcMarker
'

Global Const VtMarkerStyleDash = 0
Global Const VtMarkerStylePlus = 1
Global Const VtMarkerStyleX = 2
Global Const VtMarkerStyleStar = 3
Global Const VtMarkerStyleCircle = 4
Global Const VtMarkerStyleSquare = 5
Global Const VtMarkerStyleDiamond = 6
Global Const VtMarkerStyleUpTriangle = 7
Global Const VtMarkerStyleDownTriangle = 8
Global Const VtMarkerStyleFilledCircle = 9
Global Const VtMarkerStyleFilledSquare = 10
Global Const VtMarkerStyleFilledDiamond = 11
Global Const VtMarkerStyleFilledUpTriangle = 12
Global Const VtMarkerStyleFilledDownTriangle = 13

Global Const VtMarkerStyleCount = 14

Type VtcMarker
  visible As Integer
  style As Integer
  pen As VtcPen
  width As Single
End Type
```

```
'---------------------------------------------------------------
' VtcInfiniteLight
'
Type VtcInfiniteLight
   direction As VtcCoor3
   intensity As Single
End Type

'---------------------------------------------------------------
' VtProjectionType
'
Global Const VtProjectionTypePerspective = 0
Global Const VtProjectionTypeOblique = 1
Global Const VtProjectionTypeOrthogonal = 2

'---------------------------------------------------------------
' VtSmoothingType
'
Global Const VtSmoothingTypeNone = 0
Global Const VtSmoothingTypeQuadraticBSpline = 1
Global Const VtSmoothingTypeCubicBSpline = 2

'---------------------------------------------------------------
' VtcTextLayout
'
Type VtcTextLayout
        horizontalAlignment As Integer
        verticalAlignment As Integer
        orientation As Integer
        wordWrap As Integer
End Type

'---------------------------------------------------------------
'---------------------------------------------------------------
' from vtchdefs.h
'---------------------------------------------------------------
'---------------------------------------------------------------

'---------------------------------------------------------------
' chart types
'
Global Const VtChChartType3dBar = 0
Global Const VtChChartType2dBar = 1
Global Const VtChChartType3dLine = 2
Global Const VtChChartType2dLine = 3
Global Const VtChChartType3dArea = 4
Global Const VtChChartType2dArea = 5
Global Const VtChChartType3dStep = 6
Global Const VtChChartType2dStep = 7
Global Const VtChChartType3dCombination = 8
Global Const VtChChartType2dCombination = 9
Global Const VtChChartType3dHorizontalBar = 10
Global Const VtChChartType2dHorizontalBar = 11
Global Const VtChChartType3dClusteredBar = 12
Global Const VtChChartType3dPie = 13
Global Const VtChChartType2dPie = 14
Global Const VtChChartType3dDoughnut = 15
Global Const VtChChartType2dXY = 16
Global Const VtChChartType2dPolar = 17
Global Const VtChChartType2dRadar = 18
Global Const VtChChartType2dBubble = 19
Global Const VtChChartType2dHiLo = 20
Global Const VtChChartType2dGantt = 21
Global Const VtChChartType3dGantt = 22

'---------------------------------------------------------------
' series method types
```

```
'
Global Const VtChSeriesType3dBar = 0
Global Const VtChSeriesType2dBar = 1
Global Const VtChSeriesType3dHorizontalBar = 2
Global Const VtChSeriesType2dHorizontalBar = 3
Global Const VtChSeriesType3dClusteredBar = 4
Global Const VtChSeriesType3dLine = 5
Global Const VtChSeriesType2dLine = 6
Global Const VtChSeriesType3dArea = 7
Global Const VtChSeriesType2dArea = 8
Global Const VtChSeriesType3dStep = 9
Global Const VtChSeriesType2dStep = 10
Global Const VtChSeriesType2dXY = 11
Global Const VtChSeriesType2dPolar = 12
Global Const VtChSeriesType2dRadarLine = 13
Global Const VtChSeriesType2dRadarArea = 14
Global Const VtChSeriesType2dBubble = 15
Global Const VtChSeriesType2dHiLo = 16
Global Const VtChSeriesType2dHLC = 17
Global Const VtChSeriesType2dHLCRight = 18
Global Const VtChSeriesType2dOHLC = 19
Global Const VtChSeriesType2dOHLCBar = 20
Global Const VtChSeriesType2dGantt = 21
Global Const VtChSeriesType3dGantt = 22
Global Const VtChSeriesType3dPie = 23
Global Const VtChSeriesType2dPie = 24
Global Const VtChSeriesType3dDoughnut = 25
Global Const VtChSeriesType2dDates = 26

'------------------------------------------------------------
' chart part types
'
Global Const VtChPartTypeChart = 0
Global Const VtChPartTypeTitle = 1
Global Const VtChPartTypeFootnote = 2
Global Const VtChPartTypeLegend = 3
Global Const VtChPartTypePlot = 4
Global Const VtChPartTypeSeries = 5
Global Const VtChPartTypeSeriesLabel = 6
Global Const VtChPartTypePoint = 7
Global Const VtChPartTypePointLabel = 8
Global Const VtChPartTypeAxis = 9
Global Const VtChPartTypeAxisLabel = 10
Global Const VtChPartTypeAxisTitle = 11

'------------------------------------------------------------
' chart location
'
Global Const VtChLocationTypeTopLeft = 0
Global Const VtChLocationTypeTop = 1
Global Const VtChLocationTypeTopRight = 2
Global Const VtChLocationTypeLeft = 3
Global Const VtChLocationTypeRight = 4
Global Const VtChLocationTypeBottomLeft = 5
Global Const VtChLocationTypeBottom = 6
Global Const VtChLocationTypeBottomRight = 7
Global Const VtChLocationTypeCustom = 8


'------------------------------------------------------------
' VtChLabelComponents
'
Global Const VtChLabelComponentValue = &H1
Global Const VtChLabelComponentPercent = &H2
Global Const VtChLabelComponentSeriesName = &H4
Global Const VtChLabelComponentDataPointName = &H8


'------------------------------------------------------------
' VtChLabelLineStyle
'
Global Const VtChLabelLineStyleNone = 0
Global Const VtChLabelLineStyleStraight = 1
```

```
Global Const VtChLabelLineStyleBent = 2

'---------------------------------------------------------------------------
' VtChLabelLocationType
'
Global Const VtChLabelLocationTypeNone = 0
Global Const VtChLabelLocationTypeAbovePoint = 1
Global Const VtChLabelLocationTypeBelowPoint = 2
Global Const VtChLabelLocationTypeCenter = 3
Global Const VtChLabelLocationTypeBase = 4
Global Const VtChLabelLocationTypeInside = 5
Global Const VtChLabelLocationTypeOutside = 6
Global Const VtChLabelLocationTypeLeft = 7
Global Const VtChLabelLocationTypeRight = 8

'---------------------------------------------------------------------------
' VtChSubPlotLabelLocationType
'       Describes the subchart label position
'
Global Const VtChSubPlotLabelLocationTypeNone = 0
Global Const VtChSubPlotLabelLocationTypeAbove = 1
Global Const VtChSubPlotLabelLocationTypeBelow = 2
Global Const VtChSubPlotLabelLocationTypeCenter = 3

'---------------------------------------------------------------------------
' VtChPieWeightBasis
'       Describes the basis of weighting for pie charts
'
Global Const VtChPieWeightBasisNone = 0      ' all pies are the same size
Global Const VtChPieWeightBasisTotal = 1     ' pie size based on % of total of largest pie total
Global Const VtChPieWeightBasisSeries = 2    ' pie size ratios set by first series values

'---------------------------------------------------------------------------
' VtChPieWeightStyle
'       Describes how the weighting factor is applied.
'
Global Const VtChPieWeightStyleArea = 0
Global Const VtChPieWeightStyleDiameter = 1

'---------------------------------------------------------------------------
' VtChAxisId
'   Describes the axis Id
'
Global Const VtChAxisIdX = 0
Global Const VtChAxisIdY = 1
Global Const VtChAxisIdY2 = 2
Global Const VtChAxisIdZ = 3

'---------------------------------------------------------------------------
' VtChDateIntervalType
'       Describes the types of date intervals for date axes
'
Global Const VtChDateIntervalTypeNone = 0          ' no interval
Global Const VtChDateIntervalTypeDaily = 1         ' each day
Global Const VtChDateIntervalTypeWeekly = 2        ' each week
Global Const VtChDateIntervalTypeSemimonthly = 3   ' 1st and 15th each month
Global Const VtChDateIntervalTypeMonthly = 4       ' each month
Global Const VtChDateIntervalTypeYearly = 5        ' each year

'---------------------------------------------------------------------------
' VtChScaleType
'       Describes the type of scaling for value or polar axes
'
Global Const VtChScaleTypeLinear = 0
Global Const VtChScaleTypeLogarithmic = 1
Global Const VtChScaleTypePercent = 2
```

```
'-----------------------------------------------------------------------
' VtChPercentAxisBasis
'    Describes the type of percent axis scaling
'
Global Const VtChPercentAxisBasisMaxChart = 0      ' % of chart's maximum value
Global Const VtChPercentAxisBasisMaxRow = 1        ' % of division maximum value
Global Const VtChPercentAxisBasisMaxColumn = 2     ' % of series maximum value
Global Const VtChPercentAxisBasisSumChart = 3      ' % of chart's sum
Global Const VtChPercentAxisBasisSumRow = 4        ' % of division sum
Global Const VtChPercentAxisBasisSumColumn = 5     ' % of series sum


'-----------------------------------------------------------------------
' VtChAxisTickStyle
'    Describes the type of axis tick
'
Global Const VtChAxisTickStyleNone = 0
Global Const VtChAxisTickStyleCenter = 1
Global Const VtChAxisTickStyleInside = 2
Global Const VtChAxisTickStyleOutside = 3


'-----------------------------------------------------------------------
' Series statistics flags
'
Global Const VtChStatsMinimum = &H1
Global Const VtChStatsMaximum = &H2
Global Const VtChStatsMean = &H4
Global Const VtChStatsStddev = &H8
Global Const VtChStatsRegression = &H10


'-----------------------------------------------------------------------
' Printing information

' print scale types
Global Const VtPrintScaleTypeActual = 0
Global Const VtPrintScaleTypeFitted = 1
Global Const VtPrintScaleTypeStretched = 2

' print orientations
Global Const VtPrintOrientationPortrait = 0
Global Const VtPrintOrientationLandscape = 1

Type VtcChPrintInformation
        scaleType As Integer
        orientation As Integer

        topMargin As Single
        bottomMargin As Single
        leftMargin As Single
        rightMargin As Single

        centerHorizontally As Integer
        centerVertically As Integer

        monochrome As Integer
        layoutForPrinter As Integer
End Type


'-----------------------------------------------------------------------
' events
'
' mouse flags
'
Global Const VtChMouseFlagShiftKeyDown = &H4
Global Const VtChMouseFlagControlKeyDown = &H8
```

```
'---------------------------------------------------------------
' Formula One Link modes
Global Const VtChSsLinkModeOff = 0
Global Const VtChSsLinkModeOn = 1
Global Const VtChSsLinkModeAutoParse = 2

'---------------------------------------------------------------
' VtDcType
Global Const VtDcTypeNull = 0
Global Const VtDcTypeDisplay = 1
Global Const VtDcTypePrinter = 2
Global Const VtDcTypeMetafile = 3

'---------------------------------------------------------------
' VtTextLengthType
Global Const VtTextLengthTypeVirtual = 0
Global Const VtTextLengthTypeDevice = 1

'---------------------------------------------------------------
' VtChDrawMode
Global Const VtChDrawModeDraw = 0
Global Const VtChDrawModeBlit = 1

'---------------------------------------------------------------
'---------------------------------------------------------------


'---------------------------------------------------------------
' DLL methods

Declare Function VtChGetDLLVersion Lib "VTCHDLL.DLL" (major As Integer, minor As Integer) As Long

'---------------------------------------------------------------
' chart methods

Declare Function VtChLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long) As Long

Declare Function VtChDraw Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal hdc As Integer, ByVal hdcType As Integer,
rect As VtcLRect, ByVal layout As Integer, ByVal stretch As Integer) As Long


Declare Function VtChGetBitmap Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal options As Integer, handle As Integer) As
Long

Declare Function VtChGetMetaFile Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal options As Integer, handle As Integer,
xSize As Long, ySize As Long) As Long


'---------------------
' user interface support
'

Declare Function VtChActivateFormatMenu Lib "VTCHDLL.DLL" (ByVal hChart As Long, mPoint As VtcLCoor) As Long

Declare Function VtChGetAllowUserChanges Lib "VTCHDLL.DLL" (ByVal hChart As Long, allowUserChanges As Integer) As
Long

Declare Function VtChSetAllowUserChanges Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal allowUserChanges As
Integer) As Long

Declare Function VtChGetDrawMode Lib "VTCHDLL.DLL" (ByVal hChart As Long, drawMode As Integer) As Long

Declare Function VtChSetDrawMode Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal drawMode As Integer) As Long

Declare Function VtChGetTextLengthType Lib "VTCHDLL.DLL" (ByVal hChart As Long, textLengthType As Integer) As Long

Declare Function VtChSetTextLengthType Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal textLengthType As Integer) As
Long
```

```
Declare Function VtChEditCopy Lib "VTCHDLL.DLL" (ByVal hChart As Long) As Long

Declare Function VtChEditPaste Lib "VTCHDLL.DLL" (ByVal hChart As Long) As Long

'''''''
```

```
'''''''
```

```
Declare Function VtChSetRowLabel Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal row As Integer, ByVal labelIndex As Integer, ByVal label As String) As Long

Declare Function VtChSetRowLabelCount Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal count As Integer) As Long

Declare Function VtChSetSize Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal rowLabelCount As Integer, ByVal columnLabelCount As Integer, ByVal dataRowCount As Integer, ByVal dataColumnCount As Integer) As Long



' Chart title functions
'
Declare Function VtChGetTitle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal title As String, ByVal bufferSize As Integer) As Long

Declare Function VtChGetTitleBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChGetTitleFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtChGetTitleLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, visible As Integer, locationType As Integer, location As VtcRect) As Long

Declare Function VtChGetTitleTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


Declare Function VtChSetTitle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal title As String) As Long

Declare Function VtChSetTitleBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChSetTitleFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtchSetTitleLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal visible As Integer, ByVal locationType As Integer, location As VtcRect) As Long

Declare Function VtChSetTitleTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


' Chart footnote functions
'
Declare Function VtChGetFootnote Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal footnote As String, ByVal bufferSize As Integer) As Long

Declare Function VtChGetFootnoteBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChGetFootnoteFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtChGetFootnoteLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, visible As Integer, locationType As Integer, location As VtcRect) As Long

Declare Function VtChGetFootnoteTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


Declare Function VtChSetFootnote Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal footnote As String) As Long

Declare Function VtChSetFootnoteBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChSetFootnoteFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtChSetFootnoteLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal visible As Integer, ByVal locationType As Integer, location As VtcRect) As Long

Declare Function VtChSetFootnoteTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


' Chart legend functions
'
```

```
Declare Function VtChGetLegendBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChGetLegendFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtChGetLegendLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, visible As Integer, locationType As
Integer, location As VtcRect) As Long

Declare Function VtChGetLegendTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


Declare Function VtChSetLegendBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChSetLegendFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, aFont As VtcFont) As Long

Declare Function VtChSetLegendLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal visible As Integer, ByVal
locationType As Integer, location As VtcRect) As Long

Declare Function VtChSetLegendTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, layout As VtcTextLayout) As Long


'''''''
```

```
Declare Function VtChSetAxisValueScale Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal axis As Integer, ByVal index As
Integer, ByVal auto As Integer, ByVal min As Double, ByVal max As Double, ByVal majorDivs%, ByVal minorDivs%) As Long

'''''''
```

```
' Chart plot functions
'

Declare Function VtChGetAngleUnits Lib "VTCHDLL.DLL" (ByVal hChart As Long, angleUnits As Integer) As Long

Declare Function VtChGetBaseAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, baseHeight As Single, brush As
VtcBrush, pen As VtcPen) As Long

Declare Function VtChGetBubbleAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, maxBubbleToAxisRatio As Single)
As Long

Declare Function VtChGetDoughnutAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, interiorRatio As Single, sides As
Integer) As Long

Declare Function VtChGetWallAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, wallWidth As Single, brush As
VtcBrush, pen As VtcPen) As Long

Declare Function VtChGetPieAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, thicknessRatio As Single, topRadiusRatio
As Single) As Long

Declare Function VtChGetPlotBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChGetPlotClockwise Lib "VTCHDLL.DLL" (ByVal hChart As Long, clockwise As Integer) As Long

Declare Function VtChGetPlotGaps Lib "VTCHDLL.DLL" (ByVal hChart As Long, barGap As Single, xGap As Single, zGap As
Single) As Long

Declare Function VtChGetPlotLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, custom As Integer, location As VtcRect)
As Long

Declare Function VtChGetPlotRatios Lib "VTCHDLL.DLL" (ByVal hChart As Long, widthToHeightRatio As Single,
depthToHeightRatio As Single) As Long

Declare Function VtChGetPlotSort Lib "VTCHDLL.DLL" (ByVal hChart As Long, sort As Integer) As Long

Declare Function VtChGetPlotWeighting Lib "VTCHDLL.DLL" (ByVal hChart As Long, basis As Integer, style As Integer) As
Long

Declare Function VtChGetPerspective Lib "VTCHDLL.DLL" (ByVal hChart As Long, perspective As VtcCoor3) As Long

Declare Function VtChGetProjection Lib "VTCHDLL.DLL" (ByVal hChart As Long, projection As Integer) As Long

Declare Function VtChGetScaleAngle Lib "VTCHDLL.DLL" (ByVal hChart As Long, scaleAngle As Single) As Long

Declare Function VtChGetStartingAngle Lib "VTCHDLL.DLL" (ByVal hChart As Long, startingAngle As Single) As Long

Declare Function VtChGetSubPlotLabelPosition Lib "VTCHDLL.DLL" (ByVal hChart As Long, location As Integer) As Long

Declare Function VtChGet3DView Lib "VTCHDLL.DLL" (ByVal hChart As Long, rotation As Single, elevation As Single) As
Long


Declare Function VtChSetAngleUnits Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal angleUnits As Integer) As Long

Declare Function VtChSetBaseAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal baseHeight As Single, brush As
VtcBrush, pen As VtcPen) As Long

Declare Function VtChSetBubbleAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal maxBubbleToAxisRatio As
Single) As Long

Declare Function VtChSetDoughnutAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal interiorRatio As Single,
ByVal sides As Integer) As Long

Declare Function VtChSetWallAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal wallWidth As Single, brush As
VtcBrush, pen As VtcPen) As Long

Declare Function VtChSetPieAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal thicknessRatio As Single, ByVal
topRadiusRatio As Single) As Long

Declare Function VtChSetPlotBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, aBackdrop As VtcBackdrop) As Long

Declare Function VtChSetPlotClockwise Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal clockwise As Integer) As Long
```

```
Declare Function VtChSetPlotGaps Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal barGap As Single, ByVal xGap As
Single, ByVal zGap As Single) As Long

Declare Function VtchSetPlotLocation Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal custom As Integer, location As
VtcRect) As Long

Declare Function VtChSetPlotRatios Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal widthToHeightRatio As Single, ByVal
depthToHeightRatio As Single) As Long

Declare Function VtChSetPlotSort Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal sort As Integer) As Long

Declare Function VtChSetPlotWeighting Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal basis As Integer, ByVal style As
Integer) As Long

Declare Function VtChSetPerspective Lib "VTCHDLL.DLL" (ByVal hChart As Long, perspective As VtcCoor3) As Long

Declare Function VtChSetProjection Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal projection As Integer) As Long

Declare Function VtChSetScaleAngle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal scaleAngle As Single) As Long

Declare Function VtChSetStartingAngle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal startingAngle As Single) As Long

Declare Function VtChSetSubPlotLabelPosition Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal location As Integer) As
Long

Declare Function VtChSet3DView Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal rotation As Single, ByVal elevation As
Single) As Long

'-------------------------------------------------------
'    plot lighting properties
'

Declare Function VtChAddLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, light As VtcInfiniteLight) As Long

Declare Function VtChGetLightCount Lib "VTCHDLL.DLL" (ByVal hChart As Long, count As Integer) As Long

Declare Function VtChGetLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal index As Integer, light As VtcInfiniteLight)
As Long

Declare Function VtChRemoveLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, light As VtcInfiniteLight) As Long

Declare Function VtChRemoveLightByIndex Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal index As Integer) As Long


Declare Function VtChGetAmbientLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, ambientLightIntesity As Single) As Long

Declare Function VtChGetEdgeLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, useEdgeLight As Integer, edgeLightIntensity
As Single) As Long

Declare Function VtChSetAmbientLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal ambientLightIntesity As Single)
As Long

Declare Function VtChSetEdgeLight Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal useEdgeLight As Integer, ByVal
edgeLightIntensity As Single) As Long


'-------------------------------------------------------
'    chart series properties
'

Declare Function VtChGetSeriesBarAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, sides As
Integer, topRatio As Single) As Long

Declare Function VtChGetSeriesGuideLinePen Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, pen As
VtcPen) As Long

Declare Function VtChGetSeriesHiLoAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer,
gainColor As VtcColor, lossColor As VtcColor) As Long

Declare Function VtChGetSeriesMarkersAndLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer,
autoMarkers As Integer, showMarkers As Integer, showLines As Integer) As Long
```

Declare Function VtChGetSeriesType Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, seriesType As Integer) As Long

Declare Function VtChGetSeriesPen Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, pen As VtcPen) As Long

Declare Function VtChGetSeriesPosition Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, excluded As Integer, hidden As Integer, order As Integer, stackOrder As Integer) As Long

Declare Function VtChGetSeriesSecondaryAxis Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, secondaryAxis As Integer) As Long

Declare Function VtChGetSeriesSmoothing Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, smoothingType As Integer, smoothingFactor As Integer) As Long

Declare Function VtChGetSeriesGuideLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal axisId As Integer, showGuideLines As Integer) As Long

Declare Function VtChSetSeriesBarAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal sides As Integer, ByVal topRatio As Single) As Long

Declare Function VtChSetSeriesGuideLinePen Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, pen As VtcPen) As Long

Declare Function VtChSetSeriesHiLoAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, gainColor As VtcColor, lossColor As VtcColor) As Long

Declare Function VtChSetSeriesMarkersAndLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal autoMarkers As Integer, ByVal showMarkers As Integer, ByVal showLines As Integer) As Long

Declare Function VtChSetSeriesType Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal seriesType As Integer) As Long

Declare Function VtChSetSeriesPen Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, pen As VtcPen) As Long

Declare Function VtChSetSeriesPosition Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal excluded As Integer, ByVal hidden As Integer, ByVal order As Integer, ByVal stackOrder As Integer) As Long

Declare Function VtChSetSeriesSecondaryAxis Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal secondaryAxis As Integer) As Long

Declare Function VtChSetSeriesSmoothing Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal smoothingType As Integer, ByVal smoothingFactor As Integer) As Long

Declare Function VtChSetSeriesGuideLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal axisId As Integer, ByVal showGuideLines As Integer) As Long

---

series stat lines properties

Declare Function VtChGetSeriesStatLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, flags As Integer) As Long

Declare Function VtChGetSeriesStatLinesAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, lineColor As VtcColor, lineWidth As Single) As Long

Declare Function VtChGetSeriesStatLinesStyle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal statType As Integer, style As Integer) As Long

Declare Function VtChSetSeriesStatLines Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal flags As Integer) As Long

Declare Function VtChSetSeriesStatLinesAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, lineColor As VtcColor, ByVal lineWidth As Single) As Long

Declare Function VtChSetSeriesStatLinesStyle Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal statType As Integer, ByVal style As Integer) As Long

```
'-----------------------------------------------------------
' chart series label properties
'

Declare Function VtChGetSeriesLabel Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal buffer As
String, ByVal bufferSize As Integer) As Long

Declare Function VtChGetSeriesLabelAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, location
As Integer, lineStyle As Integer, offset As VtcCoor) As Long

Declare Function VtChGetSeriesLabelBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer,
aBackdrop As VtcBackdrop) As Long

Declare Function VtChGetSeriesLabelFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, aFont As
VtcFont) As Long

Declare Function VtChGetSeriesLabelTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, layout
As VtcTextLayout) As Long

Declare Function VtChGetSeriesLegendText Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
buffer As String, ByVal bufferSize As Integer) As Long


Declare Function VtChSetSeriesLabel Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal text As
String) As Long

Declare Function VtChSetSeriesLabelAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
location As Integer, ByVal lineStyle As Integer, offset As VtcCoor) As Long

Declare Function VtChSetSeriesLabelBackdrop Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer,
aBackdrop As VtcBackdrop) As Long

Declare Function VtChSetSeriesLabelFont Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, aFont As
VtcFont) As Long

Declare Function VtChSetSeriesLabelTextLayout Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, layout
As VtcTextLayout) As Long

Declare Function VtChSetSeriesLegendText Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal text
As String) As Long


'-----------------------------------------------------------
' point properties
'

Declare Function VtChGetPointAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
dataPoint As Integer, brush As VtcBrush, edgePen As VtcPen, marker As VtcMarker) As Long

Declare Function VtChGetPointPicture Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal dataPoint
As Integer, picture As VtcPicture) As Long

Declare Function VtChGetPointOffset Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal dataPoint
As Integer, offset As Single) As Long


Declare Function VtChResetCustomPoint Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
dataPoint As Integer) As Long

Declare Function VtChResetCustomPointLabel Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
dataPoint As Integer) As Long


Declare Function VtChSetPointAttributes Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal
dataPoint As Integer, brush As VtcBrush, edgePen As VtcPen, marker As VtcMarker) As Long

Declare Function VtChSetPointPicture Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal dataPoint
As Integer, picture As VtcPicture) As Long

Declare Function VtChSetPointOffset Lib "VTCHDLL.DLL" (ByVal hChart As Long, ByVal series As Integer, ByVal dataPoint
As Integer, ByVal offset As Single) As Long
```

```
'''''''
```